



(51) International Patent Classification:

G06N 20/00 (2019.01) G06V 20/69 (2022.01)
G06V 10/40 (2022.01) G16B 20/00 (2019.01)
G06V 10/82 (2022.01)

(21) International Application Number:

PCT/US2024/053235

(22) International Filing Date:

28 October 2024 (28.10.2024)

(25) Filing Language:

English

(26) Publication Language:

English

(30) Priority Data:

63/546,385	30 October 2023 (30.10.2023)	US
2036275	15 November 2023 (15.11.2023)	NL
63/618,660	08 January 2024 (08.01.2024)	US
2036929	30 January 2024 (30.01.2024)	NL

(71) Applicant: **DEEPCELL, INC.** [US/US]; 3925 Bohannon Drive, Suite 100, Menlo Park, California 94025 (US).

(72) Inventors: **CARELLI, Ryan C.**; 3925 Bohannon Drive, Suite 100, Menlo Park, California 94025 (US). **LUENGO HENDRIKS, Cristian L.**; 3925 Bohannon Drive, Suite 100, Menlo Park, California 94025 (US). **ZHANG, Senzeyu**; 3925 Bohannon Drive, Suite 100, Menlo Park, California 94025 (US). **JACOBS, Kevin B.**; 3925 Bohannon Drive, Suite 100, Menlo Park, California 94025 (US). **KIA, Amirali**; 3925 Bohannon Drive, Suite 100, Menlo Park, California 94025 (US). **SALEK, Mahyar**; 3925 Bohannon Drive, Suite 100, Menlo Park, California 94025 (US).

(74) Agent: **KO, Seung Hyeon** et al.; WILSON SONSINI GOODRICH & ROSATI, 650 Page Mill Road, Palo Alto, California 94304 (US).

(81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CV, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IQ, IR, IS, IT, JM, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, MG, MK, MN, MU, MW, MX, MY, MZ, NA,

(54) Title: SYSTEMS AND METHODS FOR MORPHOMETRIC ANALYSIS

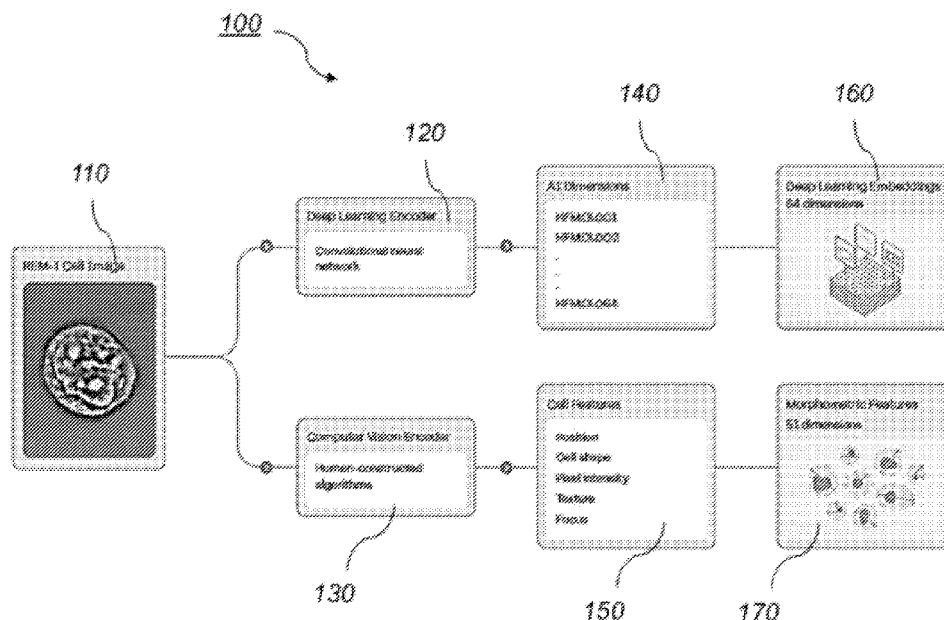


FIG. 1

(57) Abstract: Examples of the present disclosure provide systems and methods for assessing image data that includes extracting, using a Deep Learning (DL) model and image data of a plurality of cells, a vector for a cell of the plurality of cells, the vector comprising a set of machine-learning (ML)-based features and a set of cell morphometric features extracted using a computer vision model; and generating, using the DL model and using the set of ML-based features, a plurality of DL embeddings orthogonal to each other.

NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO,
RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, ST, SV, SY, TH,
TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, WS,
ZA, ZM, ZW.

- (84) Designated States** (*unless otherwise indicated, for every kind of regional protection available*): ARIPO (BW, CV, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SC, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, ME, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

- *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))*

Published:

- *with international search report (Art. 21(3))*

SYSTEMS AND METHODS FOR MORPHOMETRIC ANALYSIS**CROSS-REFERENCE**

[0001] This application claims the benefit of U.S. Provisional Application No. 63/546,385, filed October 30, 2023, U.S. Provisional Application No. 63/618, 660, filed January 8, 2024, Application Serial No. NL 2036275, filed November 15, 2023, and Application Serial No. NL 2036929, filed January 30, 2024, each of which application is incorporated herein by reference.

BACKGROUND

[0002] Analysis of a cell (e.g., determination of a type or a state of the cell) can be accomplished by examining, for example, one or more images of the cell that is tagged (e.g., stained with a polypeptide, such as an antibody, against a target protein of interest within the cell; with a polynucleotide against a target gene of interest within the cell; with probes to analyze gene expression profile of the cell via polymerase chain reaction; or with a small molecule substrate that is modified by the target protein) or sequencing data of the cell (e.g., gene fragment analysis, whole-genome sequencing, whole-exome sequencing, RNA-seq, etc.). Such methods can be used to identify cell type (e.g., stem cell or differentiated cell) or cell state (e.g., healthy or disease state). Such methods can require treatment of the cell (e.g., antibody staining, cell lysis or sequencing etc.) that can be time-consuming and/or costly.

SUMMARY

[0003] In view of the foregoing, disclosed herein are alternative methods and systems for analyzing cells (e.g., previously uncharacterized or unknown cells). For example, recognized herein is a need for method for analyzing cells without pretreatment of the cells to, e.g., tag a target protein or gene of interest in the cells, obtain sequencing data of the cells, etc.

[0004] Example 1. A method of processing, comprising:

extracting, using a Deep Learning (DL) model, a set of machine learning (ML)-based features from a cell image;

generating, using the DL model and the set of ML-based features, a plurality of DL embeddings orthogonal to each other;

extracting, using a computer vision model, a set of cell morphometric features from the cell image, the cell morphometric features being orthogonal to the plurality of DL embeddings; and

generating a plurality of morphometric predictive embeddings using the set of ML-based features and the set of cell morphometric features.

[0005] Example 2. The method of example 1, wherein the generating the plurality of morphometric predictive embeddings comprises:

generating, using the computer vision model and the set of the cell morphometric features, a plurality of morphometric embeddings; and

concatenating, using the DL model, the plurality of morphometric embeddings and the plurality of DL embeddings to a plurality of predictive multidimensional vectors comprising predictive probabilities morphometric features.

[0006] Example 3. The method of example 1, wherein the DL model is trained using a loss function comprising invariance, variance, covariance, morphometric decorrelation terms, or combinations thereof.

[0007] Example 4. The method of example 1, further comprising: predicting, using the DL model and the plurality of morphometric predictive embeddings, a plurality of morphometric blob features using at least a correlation coefficient greater than approximately 0.9.

[0008] Example 5. The method of example 1, further comprising: predicting, using the DL model and the plurality of morphometric predictive embeddings, a plurality of morphometric blob features using at least a correlation coefficient ranging between approximately 0.85 and approximately 0.95.

[0009] Example 6. The method of example 1, wherein the set of cell morphometric features comprises a plurality of blob features.

[0010] Example 7. The method of example 1, wherein the generating the plurality of morphometric predictive embeddings is in a high throughput setting.

[0011] Example 8. The method of example 1, wherein the plurality of DL embeddings comprises cell morphology information independent from a fixed set of rule-based morphometric features.

[0012] Example 9. The method of example 1, wherein the plurality of DL embeddings comprises cell morphology data orthogonal to a fixed set of rule-based morphometric features.

[0013] Example 10. The method of example 1, further comprising: separating, using the DL model, a plurality of de-correlated DL embeddings and the plurality of morphometric predictive embeddings into different dimensions.

[0014] Example 11. The method of example 1, wherein the DL model is a self-supervised machine learning (SSL) system, the method further comprising: de-correlating, using the DL model, the set of morphometric features from the DL embeddings such that the DL model is trained

to acquire information not covered using the computer vision model.

[0015] Example 12. The method of example 1, wherein the cell image comprises a label free image.

[0016] Example 13. The method of example 1, further comprising: hosting the DL model and the computer vision model in a cloud computing environment.

[0017] Example 14. The method of example 1, wherein the method is performed in a cloud computing environment.

[0018] Example 15. The method of example 1, further comprising: generating an instruction to sort a cell of the cell image based on the plurality of morphometric predictive embeddings.

[0019] Example 16. The method of example 1, further comprising: sorting, using the plurality of morphometric predictive embeddings, a cell of the cell image; and

feeding data from the sorting back to the DL model in order to train the DL model for future generating of the plurality of DL embeddings.

[0020] Example 17. A method for assessing image data, comprising:

extracting, using a Deep Learning (DL) model and image data of a plurality of cells, a vector for a cell of the plurality of cells, the vector comprising a set of machine-learning (ML)-based features and a set of cell morphometric features extracted using a computer vision model; and

generating, using the DL model and using the set of ML-based features, a plurality of DL embeddings orthogonal to each other.

[0021] Example 18. The method of example 17, wherein the DL model is trained using a loss function comprising invariance, variance, covariance, morphometric decorrelation terms, or a combination thereof.

[0022] Example 19. The method of example 17, further comprising: generating, using the DL model, a plurality of morphometric predictive embeddings using the set of ML-based features and the set of cell morphometric features.

[0023] Example 20. The method of example 19, wherein the generating the plurality of morphometric predictive embeddings comprises:

generating, using the computer vision model and at least the set of the cell morphometric features, a plurality of morphometric embeddings; and

concatenating, using the DL model, the plurality of morphometric embeddings and the plurality of DL embeddings to a plurality of predictive multidimensional vectors comprising predictive probabilities morphometric features.

[0024] Example 21. The method of example 19, further comprising: predicting, using the DL model and at least the plurality of morphometric predictive embeddings, a plurality of morphometric blob features using a correlation ranging between approximately 0.85 to approximately 0.95.

[0025] Example 22. The method of example 17, wherein the plurality of DL embeddings comprise cell morphology information orthogonal to the set of morphometric features, and wherein the set of morphometric features are determined using at least a fixed set of rules.

[0026] Example 23. The method of example 17, further comprising: separating, using the DL model, a plurality of de-correlated DL embeddings and the plurality of morphometric predictive embeddings into different dimensions.

[0027] Example 24. The method of example 17, wherein the DL model is a self-supervised machine learning (SSL) system, the method comprising: de-correlating, using the DL model, the set of morphometric features from the plurality of DL embeddings so that the DL model is trained to acquire information not covered using the computer vision model.

[0028] Example 25. The method of example 17, wherein the image data comprises a label free image of each cell of the plurality of cells.

[0029] Example 26. The method of example 17, further comprising: hosting the DL model and the computer vision model in a cloud computing environment.

[0030] Example 27. The method of example 17, further comprising: generating an instruction to sort the plurality of cells using the plurality of DL embeddings.

[0031] Example 28. The method of example 17, further comprising: sorting, using the plurality of morphometric predictive embeddings, a cell of the cell image; and

[0032] feeding data from the sorting to the DL model in order to train the DL model for future generating of the plurality of DL embeddings.

[0033] Example 29. A system for analyzing image data, the system comprising:
at least one processor to execute instructions to perform operations comprising:

extracting, using a Deep Learning (DL) model, a set of machine learning (ML)-based features from a cell image;

generating, using the DL model and using at least the ML-based features, a plurality of DL embeddings orthogonal to each other;

extracting, using a computer vision model, a set of cell morphometric features from the cell image that are orthogonal to the plurality of DL embeddings; and

generating a plurality of morphometric predictive embeddings using the set of ML-based features and the set of cell morphometric features.

[0034] Example 30. The system of Claim 29, wherein the set of cell morphometric features comprises a plurality of blob features.

[0035] Example 31. The system of Claim 29, wherein the generating the plurality of morphometric predictive embeddings is in a high throughput setting.

[0036] Example 32. The system of Claim 29, wherein the plurality of DL embeddings comprise cell morphology data orthogonal to a fixed set of rule based morphometric features.

[0037] Example 33. The system of Claim 29, the operations further comprising: separating, using the DL model, a plurality of de-correlated DL embeddings and the plurality of morphometric predictive embeddings into different dimensions.

[0038] Example 34. The system of Claim 29, wherein the DL model is a self-supervised machine learning (SSL) system, the operations further comprising: de-correlating the set of morphometric features from the plurality of DL embeddings so that the plurality of DL model is trained to acquire information not covered using the computer vision model.

[0039] Example 35. The system of Claim 29, wherein the at least one processor is in a cloud computing environment.

[0040] Example 36. A non-transitory computer-readable medium storing instructions that, when executed by processor, cause the processor to perform operations for analyzing image data of a cell image, the operations comprising:

extracting, using a trained Deep Learning (DL) model and from image data a plurality of cells, a vector for a cell of the plurality of cells, the vector comprising a set of machine-learning (ML)-based features and a set of cell morphometric features extracted by a trained computer vision model;

generating, using the trained DL model and the ML-based features, a plurality of DL embeddings orthogonal to each other; and

extracting, using the trained computer vision model, a set of cell morphometric features from the cell image that are orthogonal to the plurality of DL embeddings.

[0041] Example 37. A cloud-based computing system, the system comprising:

at least one cloud-based processor to execute the instructions to perform operations comprising:

extracting, using a Deep Learning (DL) model, a set of machine learning (ML)-based features from a cell image;

generating, using the DL model and using at least the set of ML-based features, a plurality of DL embeddings orthogonal to each other;

extracting, using a computer vision model, a set of cell morphometric features from the cell image, the cell morphometric features being orthogonal to the plurality of DL embeddings; and

generating a plurality of morphometric predictive embeddings using the set of ML-based features and the set of cell morphometric features.

[0042] Example 38. The system of Claim 37, wherein the generating the plurality of morphometric predictive embeddings comprises:

generating, using the computer vision model and at least the set of the cell morphometric features, a plurality of morphometric embeddings; and

concatenating, using the DL model, the plurality of morphometric embeddings and the plurality of DL embeddings to a plurality of predictive multidimensional vectors comprising predictive probabilities morphometric features.

[0043] Example 39. The system of Claim 37, wherein the DL model is trained using a loss function comprising one or more invariance, variance, covariance, and morphometric decorrelation terms.

[0044] Example 40. The system of Claim 37, the operations further comprising: predicting, using the DL model and the plurality of morphometric predictive embeddings, a plurality of morphometric blob features using at least a correlation coefficient greater than approximately 0.9.

[0045] Example 41. The system of Claim 37, the operations further comprising: predicting, using the DL model and the plurality of morphometric predictive embeddings, a plurality of morphometric blob features using at least a correlation coefficient ranging between approximately 0.85 and approximately 0.95.

[0046] Example 42. The system of Claim 37, the operations further comprising: separating, using the DL model, a plurality of de-correlated DL embeddings and the plurality of morphometric predictive embeddings into different dimensions.

[0047] Example 43. The system of Claim 37, wherein the DL model is a self-supervised machine learning (SSL) system, the operations further comprising: de-correlating, using the DL model, the set of morphometric features from the DL embeddings such that the DL model is trained to acquire information not covered using the computer vision model.

[0048] Example 44. A method for cell sorting, comprising:

transporting a cell suspended in a fluid through a flow channel, wherein the flow channel is in fluid communication with a plurality of sub-channels;

capturing one or more images of the cell as the cell is transported through the flow channel;

extracting, by a Deep Learning (DL) model, a set of machine learning (ML)-based features from the one or more images;

generating, using the DL model and the set of ML-based features, a plurality of DL embeddings orthogonal to each other;

extracting, using a computer vision model, a set of cell morphometric features from the one or more images, the cell morphometric features being orthogonal to the plurality of DL embeddings;

generating a plurality of morphometric predictive embeddings using the set of ML-based features and the set of cell morphometric features; and

sorting the cell to a selected sub-channel of the plurality of sub-channels using the plurality of morphometric predictive embeddings.

[0049] Example 45. A method for cell sorting, comprising:

transporting a cell suspended in a fluid through a flow channel, wherein the flow channel is in fluid communication with a plurality of sub-channels;

capturing one or more images of the cell as the cell is transported through the flow channel;

extracting, by a Deep Learning (DL) model, a set of machine learning (ML)-based features from the one or more images;

generating, using the DL model and using the set of ML-based features, a plurality of DL embeddings orthogonal to each other; and

sorting the cell to a selected sub-channel of the plurality of sub-channels using the plurality of DL embeddings.

[0050] Example 46. A cell sorting system, comprising:

a flow channel configured to transport a cell through the flow channel;

an imaging device configured to capture one or more images of the cell as the cell is transported through the flow channel; and

a processor to execute instructions to perform operations comprising:

extracting, using a Deep Learning (DL) model, a set of machine learning (ML)-based features from a cell image;

generating, using the DL model and using at least the ML-based features, a plurality of DL embeddings orthogonal to each other;

extracting, using a computer vision model, a set of cell morphometric features from the cell image that are orthogonal to the plurality of DL embeddings;

generating a plurality of morphometric predictive embeddings using the set of ML-based features and the set of cell morphometric features; and

enabling sorting of the cell using at least one of the plurality of morphometric predictive embeddings, the plurality of DL embeddings, and the set of cell morphometric features.

[0051] Example 47. A cell sorting system, comprising:

a flow channel configured to transport a cell through the flow channel;

an imaging device configured to capture one or more images of the cell as the cell is transported through the flow channel; and

a processor to execute instructions to perform operations comprising:

extracting, using a Deep Learning (DL) model and image data of a plurality of cells, a vector for a cell of the plurality of cells, the vector comprising a set of machine-learning (ML)-based features and a set of cell morphometric features extracted using a computer vision model;

generating, using the DL model and using the set of ML-based features, a plurality of DL embeddings orthogonal to each other; and

enabling sorting of the cell using at least one of the plurality of DL embeddings and the set of cell morphometric features.

[0052] Additional aspects and advantages of the present disclosure will become readily apparent to those skilled in this art from the following detailed description, wherein only illustrative embodiments of the present disclosure are shown and described. As will be realized, the present disclosure is capable of other and different embodiments, and its several details are capable of modifications in various obvious respects, all without departing from the disclosure. Accordingly, the drawings and description are to be regarded as illustrative in nature, and not as restrictive.

BRIEF DESCRIPTION OF THE DRAWINGS

[0053] The novel features of the disclosure are set forth with particularity in the appended claims. A better understanding of the features and advantages of the present disclosure will be obtained by reference to the following detailed description that sets forth illustrative embodiments, in which the principles of the disclosure are utilized, and the accompanying drawings (also “Figure” and “FIG.” herein), of which:

[0054] FIG. 1 illustrates an example workflow of extracting features associated with cell morphology from cell images using the human foundation model, in accordance with some examples of the present disclosure.

- [0055] FIG. 2A illustrates an example interaction between a microfluidics platform, the human foundation model, and the data suite, in accordance with some examples of the present disclosure.
- [0056] FIG. 2B illustrates an example workflow from high-throughput imaging to cell characterization, classification and sorting based on cell morphology analysis, in accordance with some examples of the present disclosure.
- [0057] FIG. 3 schematically illustrates an example method for classifying a cell.
- [0058] FIG. 4 schematically illustrates, in one example, different ways of representing analysis data of image data of cells.
- [0059] FIG. 5 schematically illustrates, in one example, different representations of analysis of image data of a population of cells.
- [0060] FIG. 6 schematically illustrates, in one example, a method for a user to interact with a method for analyzing image data of cells.
- [0061] FIG. 7 schematically illustrates, in one example, a cell analysis platform for analyzing image data of one or more cells.
- [0062] FIGS. 8A-8B schematically illustrate, in one example, an example microfluidic system for sorting one or more cells.
- [0063] FIG. 9 illustrates an example training architecture of the human foundation model.
- [0064] FIG. 10 illustrates another example of the training architecture of the human foundation model shown in FIG. 9.
- [0065] FIGS. 11A and 11B show examples of morphometric features of cellular images.
- [0066] FIG. 12A illustrates cell classes, numbers of images used as training dataset to train the human foundation model, numbers of images processed by the human foundation model as test dataset, and corresponding representative cell images, in accordance with some examples of the present disclosure.
- [0067] FIG. 12B illustrates an example confusion matrix between predicted cell classes classified by the human foundation model and actual cell classes, in accordance with some examples of the present disclosure.
- [0068] FIG. 13 with views (a) to (f) schematically illustrate an example system for classifying and sorting one or more cells.
- [0069] FIG. 14 with views (a) to (e) schematically illustrate operations that can be performed in an example method.
- [0070] FIG. 15 shows, in one example, a computer system that is programmed or otherwise configured to implement methods provided herein.
- [0071] FIG. 16 illustrates an example flow of operations in a method of processing images.

- [0072] FIG. 17 illustrates an example flow of operations in a method of processing images.
- [0073] FIG. 18 illustrates, in one example, how a convolution layer may identify features in an input image.
- [0074] FIG. 19 illustrates, in one example, how a transpose convolution layer may operate.
- [0075] FIGS. 20A-20E illustrate, in one example, morphometric characteristics which may be used as dimensions in feature space for cell images.
- [0076] FIG. 21 illustrates, in one example, a method which can be used to train a neural network to approximate the process of recreating an original input after it has been degraded through the introduction of noise.
- [0077] FIG. 22 illustrates, in one example, a method which may be used to create new images using a neural network trained in a manner such as shown in FIG. 21.
- [0078] FIG. 23 illustrates, in one example, an image creation process which is guided by additional information such as a complete or partial embedding.
- [0079] FIG. 24 depicts, in one example, a process which may be used in exploring a feature space.
- [0080] FIG. 25, in one example, depicts an interface which may be used to display synthetic cell images.
- [0081] FIG. 26 depicts, in one example, hardware optimized for rendering and/or parallel operations.
- [0082] FIG. 27 depicts an example imaging system.

INCORPORATION BY REFERENCE

[0083] All publications, patents, and patent applications mentioned in this specification are herein incorporated by reference in their entirety, to the same extent as if each individual publication, patent, or patent application is specifically and individually indicated to be incorporated by reference. To the extent publications and patents or patent applications incorporated by reference contradict the disclosure contained in the specification, the specification is intended to supersede and/or take precedence over any such contradictory material.

DETAILED DESCRIPTION

[0084] While various examples provided herein have been shown and described herein, it will be obvious to those skilled in the art that such examples are provided by way of illustration only. Numerous variations, changes, and substitutions may occur to those skilled in the art without departing from the present disclosure. It should be understood that various alternatives to the

examples of the present disclosure described herein can be employed.

[0085] Unless defined otherwise, all technical and scientific terms used herein have the same meaning as commonly understood by one of ordinary skill in the art to which the present disclosure belongs. In case of conflict, the present application including the definitions will control. Also, unless otherwise required by context, singular terms shall include pluralities and plural terms shall include the singular.

[0086] Throughout this specification and the claims which follow, unless the context requires otherwise, the word “comprise,” and variations such as “comprises” and “comprising” means various components may be co-jointly employed in the methods and articles (e.g., compositions and apparatuses including device and methods). For example, the term “comprising” will be understood to imply the inclusion of any stated elements or acts but not the exclusion of any other elements or acts. In general, any of the apparatuses and methods described herein are inclusive, but all or a sub-set of the components and/or acts may alternatively be exclusive and may be expressed as “consisting of” or alternatively “consisting essentially of” the various components, acts, sub-components, or sub-acts. Furthermore, references to “one example” are not intended to be interpreted as excluding the existence of additional examples that also incorporate the recited features. The use of “including,” “comprising,” “having,” or “in which,” and variations thereof, herein is meant to encompass the items listed thereafter and equivalents thereof as well as additional items.

[0087] Whenever the term “at least,” “greater than,” or “greater than or equal to” precedes the first numerical value in a series of two or more numerical values, the term “at least,” “greater than” or “greater than or equal to” applies to each of the numerical values in that series of numerical values. For example, greater than or equal to about 1, about 2, or about 3 is equivalent to greater than or equal to about 1, greater than or equal to about 2, or greater than or equal to about 3.

[0088] Whenever the term “no more than,” “less than,” or “less than or equal to” precedes the first numerical value in a series of two or more numerical values, the term “no more than,” “less than,” or “less than or equal to” applies to each of the numerical values in that series of numerical values. For example, less than or equal to about 3, about 2, or about 1 is equivalent to less than or equal to about 3, less than or equal to about 2, or less than or equal to about 1.

[0089] The term “morphology” or “morphological characteristic” or “morphological feature” of a cell as used herein generally refers to the form, structure, and/or configuration of the cell. The term “morphometric feature” is intended to refer to a quantitative representation of a morphological feature of a cell, and in some cases the terms “morphological feature” and “morphometric feature” are used interchangeably herein. The morphology of a cell can comprise one or more aspects of a

cell's appearance, such as, for example, shape, size, arrangement, form, structure, pattern(s) of one or more internal and/or external parts of the cell, or shade (e.g., color, greyscale, etc.). Non-limiting examples of a shape of a cell can include, but are not limited to, circular, elliptic, shmoo-like, dumbbell, star-like, flat, scale-like, columnar, invaginated, having one or more concavely formed walls, having one or more convexly formed walls, prolonged, having appendices, having cilia, having angle(s), having corner(s), etc. A morphological feature of a cell can be visible with treatment of a cell (e.g., small molecule or antibody staining). In another example, the morphological feature of the cell need not require any treatment to be visualized in an image or video.

[0090] The terms “unstructured” or “unsorted,” as used interchangeably herein, generally refers to a mixture of cells (e.g., an initial mixture of cells) that is not substantially sorted (or rearranged) into separate partitions. An unstructured population of cells can comprise at least two types of cells that can be distinguished by exhibiting different properties (e.g., one or more physical properties, such as one or more different morphological characteristics as disclosed herein). The unstructured population of cells can be a random (or randomized) mixture of the at least two types of cells. The cells as disclosed herein can be viable cells. A viable cell, as disclosed herein, can be a cell that is not undergoing necrosis or a cell that is not in an early or late apoptotic state. Assays for determining cell viability can include, e.g., as using propidium iodide (PI) staining which can be detected by flow cytometry. In another example, the cells need not be viable (e.g., fixed cells).

[0091] A “viable cell” as disclosed herein can be characterized by exhibiting one or more characteristics (e.g., morphology, one or more gene expression profiles, etc.) that is substantially unaltered (or that is not substantially impacted by) by any operation or process of the methods disclosed herein (e.g., partitioning). In some examples, a characteristic of a viable cell can be a gene transcript accumulation rate, which can be characterized by a change in transcript levels of a same gene (e.g., a same endogenous gene) between mother and daughter cells over the time between cell divisions, as ascertained by single cell sequencing, polymerase chain reaction (PCR), etc.

[0092] The term, “high throughput”, when referring to a platform, system, model, and the like, means that such a platform, system, model, etc., is capable of generating an embedding for at least one image within a desired time, such as but not limited to approximately 5 ms to 30 ms. In some aspects, a high-throughput setting can also include components to process approximately 10,000 frames/sec and/or approximately 1000 images/sec while being configured to correct per-pixel variation in background offset, camera gain, and/or illumination for the processed frames. As used herein, “high-throughput” systems can require relatively low-latency.

[0093] Relative terms, such as “about,” “substantially,” or “approximately” are used to include small variations with specific numerical values (e.g., $\pm x\%$), as well as including the situation of no variation ($\pm 0\%$). In some examples, the numerical value x is less than or equal to 10 – e.g., less than or equal to 5, to 2, to 1, or smaller.

[0094] The term “real time” or “real-time,” as used interchangeably herein, generally refers to an event (e.g., an operation, a process, a method, a technique, a computation, a calculation, an analysis, an optimization, etc.) that is performed using recently obtained (e.g., collected or received) data. Examples of the event may include, but are not limited to, analysis of a one or more images of a cell to classify the cell, updating one or more deep learning algorithms (e.g., neural networks) for classification and sorting, controlling actuation of one or more valves by a sorting bifurcation, etc. In some examples, a real time event can be performed almost immediately or within a short enough time span, such as within at least about 0.0001 ms, at least about 0.0005 ms, at least about 0.001 ms, at least about 0.005 ms, at least about 0.01 ms, at least about 0.05 ms, at least about 0.1 ms, at least about 0.5 ms, at least about 1 ms, at least about 5 ms, at least about 0.01 seconds, at least about 0.05 seconds, at least about 0.1 seconds, at least about 0.5 seconds, at least about 1 second, or more. In some examples, a real time event can be performed almost immediately or within a short enough time span, such as within at most about 1 second, at most about 0.5 seconds, at most about 0.1 seconds, at most about 0.05 seconds, at most about 0.01 seconds, at most about 5 ms, at most about 1 ms, at most about 0.5 ms, at most about 0.1 ms, at most about 0.05 ms, at most about 0.01 ms, at most about 0.005 ms, at most about 0.001 ms, at most about 0.0005 ms, at most about 0.0001 ms, or less. In some examples, any of the operations of a computer processor as provided herein can be performed (e.g., automatically performed) in real-time.

[0095] As used herein, an “encoder” refers to a type of deep learning model that transforms or “encodes” an image into a vector.

[0096] As used herein, the singular forms “a,” “an,” and “the” are intended to include the plural forms as well, unless the context clearly indicates otherwise. As used herein, the term “and/or” includes any and all combinations of one or more of the associated listed items and may be abbreviated as “/”.

[0097] When used in the claims, the term “set” is one or more things which are grouped together. Similarly, “based on” indicates that one thing is determined at least in part by what it is specified as being “based on.” Where one thing is required to be exclusively determined by another thing, then that thing will be referred to as being “exclusively based on” that which it is determined by.

[0098] Spatially relative terms, such as “under,” “below,” “lower,” “over,” “upper,” and the

like, may be used herein for ease of description to describe one element or feature's relationship to another element(s) or feature(s) as illustrated in the figures. It will be understood that the spatially relative terms are intended to encompass different orientations of the device in use or operation in addition to the orientation depicted in the figures. For example, if a device in the figures is inverted, elements described as “under” or “beneath” other elements or features may then be oriented “over” the other elements or features. Thus, the term “under” may encompass both an orientation of over and under. The device may be otherwise oriented (rotated 90 degrees or at other orientations) and the spatially relative descriptors used herein interpreted accordingly. Similarly, the terms “upwardly,” “downwardly,” “vertical,” “horizontal,” and the like are used herein for the purpose of explanation only unless specifically indicated otherwise. In addition, terms such as “outer” and “inner” are used herein for purposes of description and are not intended to indicate or imply relative importance or significance.

[0099] When a feature or element is herein referred to as being “on” or “over” another feature or element, it may be directly on or over the other feature or element; or intervening features and/or elements may also be present. In other words, when a feature or element is herein referred to as being “on” or “over” another feature or element, it may be indirectly on or over the other feature or element. In contrast, when a feature or element is referred to as being “directly on” or “directly over” another feature or element, there are no intervening features or elements present.

[0100] When a feature or element is referred to as being “mounted,” “connected,” “supported,” “attached,” or “coupled” to another feature or element, it may be directly mounted, connected, supported, attached, or coupled to the other feature or element or intervening features or elements may be present. In contrast, when a feature or element is referred to as being “directly mounted,” “directly connected,” “directly supported,” “directly attached,” or “directly coupled” to another feature or element, there are no intervening features or elements present. Although described or shown with respect to one embodiment, the features and elements so described or shown may apply to other embodiments. It will also be appreciated by those skilled in the art that references to a structure or feature that is disposed “adjacent” another feature may have portions that overlap or underlie the adjacent feature.

[0101] As used herein in the specification and claims, including as used in the examples and unless otherwise expressly specified, the terms “about” or “approximately” for any numerical values or ranges indicate a suitable dimensional tolerance, or other form of reasonable expected range, that allows the part or collection of components to function for its intended purpose as described herein. More specifically, “about” or “approximately” may refer to the range of values that are within $\pm 10\%$ of the recited value, including ± 0 (e.g., “about 100” may refer to the range of

values from 90 to 110, including 90, 110, 100, and all other values within the range of 90 and 110). Any numerical values given herein include about or approximately that value unless the context indicates otherwise. For example, if the value “10” is disclosed, then “about 10” is also disclosed. Any numerical range recited herein is intended to include all sub-ranges subsumed therein. The terms “approximately” and “about” are thus utilized herein to represent the degree by which a quantitative representation may vary from a stated reference without resulting in a change in the basic function of the subject matter at issue.

[0102] The term “substantially” is also utilized herein to represent the degree by which a quantitative representation may vary from a stated reference without resulting in a change in the basic function of the subject matter at issue. The term “substantially” shall therefore be understood to include a range of conditions or results that provide a functional equivalent to an explicitly stated condition or result. For instance, if a task is “substantially complete,” the result of the task having been substantially completed is functionally equivalent to the result that may have been achieved if the task had been perfectly completed. As another non-limiting example, a component that is “substantially straight” or “substantially flat,” an apparatus including a component that is “substantially straight” or “substantially flat” may provide a result or effect that is functionally equivalent to a result or effect that may be achieved by the same apparatus including the same component in a perfectly straight or perfectly flat configuration. The range implied by the term “substantially” includes the perfect result that is within that range. Thus, the term “substantially complete” shall be read as including “perfectly complete” while also including a range of completeness that is functionally equivalent to perfectly complete. As another example, terms such as “substantially straight” and “substantially flat” shall be read as including “perfectly straight” and “perfectly flat,” respectively; while also including a range of straightness or flatness that is functionally equivalent to perfectly straight or flat, respectively. As with the terms “approximately” and “about,” the term “substantially” may indicate a suitable dimensional tolerance, or other form of reasonable expected range, that allows a part or collection of components to function for its intended purpose as described herein.

[0103] The term “perpendicular” shall be understood to include arrangements where one element (e.g., surface, feature, component, axis, etc.) defines an angle of 90 degrees with another element (e.g., surface, feature, component, axis, etc.). The term “perpendicular” shall also be understood to include arrangements where one element (e.g., surface, feature, component, axis, etc.) defines an angle of approximately 90 degrees with another element (e.g., surface, feature, component, axis, etc.).

[0104] It is also understood that when a value is disclosed that “less than or equal to” the value,

“greater than or equal to the value,” and possible ranges between values are also disclosed, as appropriately understood by the skilled artisan. For example, if the value “X” is disclosed the “less than or equal to X” as well as “greater than or equal to X” (e.g., where X is a numerical value) is also disclosed. It is also understood that throughout the application, data is provided in a number of different formats, and that this data, represents endpoints and starting points, and ranges for any combination of the data points. For example, if a particular data point “10” and a particular data point “15” are disclosed, it is understood that greater than, greater than or equal to, less than, less than or equal to, and equal to 10 and 15 are considered disclosed as well as between 10 and 15. It is also understood that each unit between two particular units are also disclosed. For example, if 10 and 15 are disclosed, then 11, 12, 13, and 14 are also disclosed.

[0105] Although the terms “first” and “second” may be used herein to describe various features/elements (including acts), these features/elements are not limited by these terms, unless the context indicates otherwise. These terms are used to distinguish one feature/element from another feature/element, and unless specifically pointed out, do not denote a certain order. Thus, a first feature/element discussed below may be termed a second feature/element, and similarly, a second feature/element discussed below may be termed a first feature/element without departing from the teachings of the present disclosure. The terms “first,” “second,” and “third,” etc. are thus used merely as labels, and are not intended to impose numerical requirements on their objects.

[0106] As used herein, the terms “system,” “apparatus,” and “device” may be read as being interchangeable with each other. A system, apparatus, and device may each include a plurality of components having various kinds of structural and/or functional relationships with each other.

[0107] The term “fluid” shall be understood to include liquids and gases, including pneumatic pressure. Similarly, “fluidic communication” shall be understood to include the communication of liquids and the communication of gases, including pneumatic pressure.

[0108] The term “morphometric feature” of a cell as used herein generally refers to the form, structure, and/or configuration of the cell. The morphometric features of a cell may comprise one or more aspects of a cell’s appearance, such as, for example, shape, size, arrangement, form, structure, pattern(s) of one or more internal and/or external parts of the cell, or shade (e.g., color, greyscale, etc.). Non-limiting examples of a shape of a cell may include, but are not limited to, circular, elliptic, dumbbell, star-like, flat, scale-like, columnar, invaginated, having one or more concavely formed walls, having one or more convexly formed walls, prolonged, having appendices, having cilia, having angle(s), having corner(s), etc. A morphometric feature of a cell may be visible with treatment of a cell (e.g., small molecule or antibody staining). In other examples, the morphometric feature of the cell may not and need not require any treatment to be

visualized in an image or video.

[0109] The terms “unstructured” or “unsorted,” as used interchangeably herein, generally refers to a mixture of cells (e.g., an initial mixture of cells) that is not substantially sorted (or rearranged) into separate partitions. An unstructured population of cells may comprise at least two types of cells that can be distinguished by exhibiting different properties (e.g., one or more physical properties, such as one or more different morphological characteristics as disclosed herein). The unstructured population of cells may be a random (or randomized) mixture of the at least two types of cells. The cells as disclosed herein may be viable cells. A viable cell, as disclosed herein, may be a cell that is not undergoing necrosis or a cell that is not in an early or late apoptotic state. In other examples, the cells may not and need not be viable (e.g., fixed cells).

[0110] The term “resilient” as used herein refers to a material property where the material has shape memory and stiffness such that it is structurally biased toward a neutral shape or structural arrangement. As an example, a resilient member may have a resilient bias toward a neutral shape or structural arrangement where the resilient member is straight along a central longitudinal axis. That same resilient member may be deformed relative to the neutral shape or structural arrangement, such as by being bent away from that central longitudinal axis, in response to a force (e.g., when a force is imparted on the resilient member, where the force has a directional component that is transverse to the central longitudinal axis). While the resilient member is being deformed relative to the neutral shape in response to the force, the resilient member may be under stress whereby the resilient property of the material of the resilient member generates a force in a direction that is opposite to the force that is causing the deformation of the resilient member. In other words, the resilient property of the material of the resilient member may impart a mechanical bias urging the resilient member back toward the neutral shape or structural arrangement. After the force causing the deformation of the resilient member is removed, the resilient bias of the material of the resilient member may cause the resilient member to return to (or at least toward) the neutral shape or structural arrangement. While the foregoing example provides a straight configuration as a neutral shape or structural arrangement, other examples of resilient members may have other kinds of neutral shapes or structural arrangements.

[0111] Morphology is an important cell property associated with identity, state, and function, but in some instances it is characterized crudely in a few standard dimensions such as diameter, perimeter, or area, or with subjective qualitative descriptions. In comparison, the present disclosure provides a method of processing that includes using a machine learning encoder to extract a set of ML-based features from a cell images, using a computer vision encoder to extract a set of cell morphometric features from the cell image, and using the set of ML-based features and the set of

cell morphometric features to generate a feature vector that represents morphology of the cell. The feature vector can be used in a variety of practical applications, e.g., in a manner such as described in the nonlimiting examples provided herein.

[0112] In one example, tumors are composed of heterogeneous assortments of cells with distinct genetic and phenotypic characteristics that may drive therapeutic resistance, immune evasion, and disease progression. The advent of single cell technologies has enabled deep profiling of individual cells within a tumor microenvironment, leading to a better understanding of tumor biology and subsequently more effective cancer treatment strategies. While profiling technologies such as flow cytometry and single cell sequencing yields insight on tumor composition, cells are sometimes no longer amenable to additional downstream studies after being subjected to antibody staining or destructive analytical processes such as cell lysis. Current sorting methods such as fluorescence-activated cell sorting (FACS) rely on a limited set of biomarkers, which cannot cover the full extent or be readily available for all distinct cell properties. Additionally, dependence on antibodies, dyes/stains, and biomarkers to denote cell identity may inadvertently create sampling bias by depleting biomarker-negative but potentially biologically interesting cell populations.

[0113] Cell morphology information has historically been used for cell and disease characterization but has been difficult to objectively and reproducibly quantify. Cell morphology in many instances are studied qualitatively through microscopes, which can be inherently slow, difficult to scale, and relies on human interpretation.

[0114] The present disclosure provides multi-dimensional morphology analysis (e.g., profiling) enabled by machine learning and computer vision morphometrics. The present disclosure has the benefit of enabling higher resolution and biological insight while reducing labor-intensive cell processing manipulations. The multi-dimensional morphology profiling and sorting of unlabeled single cells using machine learning, advanced imaging, and microfluidics can be used to assess population heterogeneity beyond biomarkers.

[0115] In some examples, the present disclosure provides a method for cell morphology analysis. In some examples, the method may combine deep learning and computer vision methods to extract features from cell images. A deep learning model used in the method may provide quantitative descriptions of cell features using one or more neural network. A computer vision model used in the method may provide a quantitative assessment of cell and biological features using discrete image analysis algorithms. The method as described herein may allow for extracting and interpreting cell morphology features with a multidimensional, unbounded, and quantitative assessment.

[0116] In some examples, the present disclosure provides a system for cell morphology

analysis. In some examples, the system may comprise a benchtop single-cell imaging and sorting system for high-dimensional morphology analysis. The system may combine label-free imaging, deep learning, computer vision morphometrics, and gentle cell sorting to leverage multidimensional single cell morphology as a quantitative readout. The system may capture high-resolution brightfield cell images, from which features (e.g., dimensional embedding vectors) can be extracted representing the morphology of the cells.

[0117] The system and method may combine label-free imaging, deep learning, computer vision morphometrics, and gentle cell sorting to harness multi-dimensional single cell morphology as a quantitative biological readout. The systems and methods disclosed herein have a variety of potential uses. For example, the combination of deep learning and computer vision morphometrics may allow cell characterization and sorting based on multi-dimensional morphometric and deep learning derived features, which can be used to identify and enrich cancer cells in heterogeneous populations. Quantitative multi-dimensional morphology information at single cell level may provide additional information to resolve cancer heterogeneity. The system and method may extract pigmentation features from the morphology profiling and based on which, assess melanoma cells. Cell populations can be characterized with specific morphological profiles have distinct molecular profiles, and morphologically distinct cells (e.g., normal vs. tumor) can be distinguished over each other.

[0118] The system and method may provide a relatively fast workflow for cell morphology analysis. For example, it may only take a few hours from preparing cell samples to generating publishable figures representing cell morphology. The systems and methods can be used in a variety of applications including but not limited to cancer research, developmental biology, cell and gene therapies, and drug and functional screening. In some examples, the systems and methods as described herein can be used in drug and Clustered Regularly Interspaced Short Palindromic Repeats (CRISPR) perturbation screening, using cell morphology as a novel biomarker for the screening. The systems and methods as described herein can be used in sample-level profiling including but not limited to heterogeneous sample evaluation and characterization, disease detection and enrichment, and sample clean-up. In other examples, the systems and methods as described herein can be used in cell-level phenotyping, including cell health status, cell state characterization, and multi-omic integration.

Human Foundation Model (HFM)

[0119] In some examples, the present disclosure may provide a human foundation model (“HFM”) for cell morphology analysis (e.g., profiling). The model may combine a deep learning

model and a computer vision model and extract cell features from cell images. In some examples, the deep learning model may process cell images as input and provide quantitative descriptions of cell features. The deep learning model may extract deep learning features that are information-rich metrics of cell morphology with powerful discriminative capabilities. The deep learning features may not be human-interpretable. The computer vision model may process cell images as input and provide morphometric features that are human-interpretable, quantitative metrics of cell morphology including cell size, shape, texture, and intensity. The morphometrics can be computationally generated using discrete computer vision algorithms. When some of the morphometrics are too computationally intensive to compute in real time, the deep learning model may overcome the limitation of the computer vision model by imputing the most computationally intensive morphometrics into the human foundation model. By combining deep learning and morphometric features, the human foundation model as described herein may provide both accuracy and interpretability in real-time feature extraction, cell classification and sorting. The human foundation model may also have strong generalization capabilities that enable hypothesis-free sample exploration and efficient generation of application-specific models.

[0120] FIG. 1 illustrates an example workflow of extracting features associated with cell morphology from cell images using the human foundation model, in accordance with some examples of the present disclosure. The human foundation model may process cell images 110 and generate features therefrom. In some examples, cells that are under analysis can be unstained, and the cell images 110 can be brightfield cell images.

[0121] In some examples, the human foundation model may comprise a deep learning model 120 and a computer vision model 130. The deep learning model 120 may comprise a deep learning encoder, for example, a convolutional neural network. The deep learning model 120 may process cell images 110 as input and extract artificial intelligence (AI) features 140 therefrom. In some examples, the AI features 140 may comprise deep learning features 160, e.g., features that are extracted using a deep learning algorithm, such as a convolutional neural network, with other nonlimiting examples being provided elsewhere herein. In some examples, the dimensions of the deep learning features can be in a range of between about 1 and about 10, between about 1 and about 20, between about 1 and about 50, between about 1 and about 80, between about 1 and about 100, between about 1 and about 200, between about 1 and about 500, between about 1 and about 800, between about 1 and about 1,000, between about 1 and about 2,000, between about 1 and about 5,000, between about 1 and about 8,000, between about 1 and about 10,000, between about 1 and about 20,000, between about 1 and about 50,000, between about 1 and about 80,000, or between about 1 and about 100,000, or any value between any of the aforementioned numbers.

Any range of numbers of the deep learning features can be contemplated, for example, for example, the number may be at least about 1 feature – e.g., at least about 5, at least about 10, at least about 50, at least about 100, at least about 500, at least about 1,000, at least about 5,000, at least about 10,000, at least about 50,000, at least about 100,000, or more, features. In another example, the number may be up to about 100,000 features – e.g., up to about 50,000 features, up to about 10,000, up to about 5,000, up to about 1,000, up to about 500, up to about 100, up to about 50, up to about 10, up to about 5, up to about 1, or smaller, features. Other suitable numbers are also possible. As one example, the deep learning model 120 may extract between about 5 and about 1000 deep learning features, e.g., between about 10 and about 500 deep learning features, e.g., between about 50 and about 100 deep learning features, from each cell image. In some examples, in a data set comprising a plurality of deep learning features of the cell(s), each feature can be referred to as a dimension (e.g., a deep learning dimension). Any range of dimensions of the deep learning features can be contemplated, for example from 1 through any number greater than about 100,000. As illustrated in FIG. 1, as one nonlimiting example, the deep learning model 120 generates about 64-dimensional deep learning features 160.

[0122] In some examples, the computer vision model 130 may comprise a computer vision encoder including human-constructed algorithms, which in some cases can be referred to as “rule-based morphometrics.” The computer vision model 130 may process cell images 110 as input and extract cell features 150 therefrom. In some examples, the cell features 150 may comprise cell position, cell shape, pixel intensity, texture, focus, or combinations thereof. The cell features 150 may comprise morphometric features 170. Nonlimiting examples of morphometric features 170 are provided below in Table 1. The dimensions of the morphometric features 170 can be in a range of between about 1 and about 10, between about 1 and about 20, between about 1 and about 50, between about 1 and about 80, between about 1 and about 100, between about 1 and about 200, between about 1 and about 500, between about 1 and about 800, between about 1 and about 1,000, between about 1 and about 2,000, between about 1 and about 5,000, between about 1 and about 8,000, between about 1 and about 10,000, between about 1 and about 20,000, between about 1 and about 50,000, between about 1 and about 80,000, or between about 1 and about 100,000, or any value between any of the aforementioned numbers. The cell features 150 may include any suitable number of morphometric features 170, for example, at least about 1 feature, at least about 5 features, at least about 10 features, at least about 50 features, at least about 100 features, at least about 500 features, at least about 1,000 features, at least about 5,000 features, at least about 10,000 features, at least about 50,000 features, and at least about 100,000 features. In some cases, in a data set comprising a plurality of computer vision features of the cell(s), each feature can be referred to

as a dimension (e.g., computer vision-based dimension). Any range of dimensions of the morphometric features can be contemplated, for example from 1 through any number greater than 100,000. As one example, the computer vision model 130 may extract between about 5 and about 1000 morphometric features, e.g., between about 10 and about 500 morphometric features, e.g., between about 50 and about 100 features, and any values in between any of the aforementioned ranges, from each cell image. As illustrated in FIG. 1, in one nonlimiting example, the computer vision model generates about 51-dimensional morphometric features 170.

[0123] In some examples, the human foundation model may encode the deep learning features 160 and morphometric features 170 into multidimensional numerical vectors representing the cell morphology. In one nonlimiting example, the 64-dimensional deep learning features 160 and 51-dimensional morphometric features 170 can be encoded into 115-dimensional embedding vectors representing the cell morphology. Further details regarding example data structures (e.g., multi-dimensional vectors) for encoding morphometric features 170 are provided further below.

[0124] In some examples, the human foundation model may generate one or more morphology maps based on deep learning features 160, morphometric features 170, or combinations thereof and in some examples based on a plurality of deep learning features and a plurality of morphometric features (e.g., based on a multi-dimensional vector that represents morphology of a cell). A cell morphology map can be a visual (e.g., graphical) representation of one or more clusters of datapoints. The cell morphology map can be a 1-dimensional (1D) representation (e.g., based on one morphological property as one parameter or dimension) or a multi-dimensional representation, such as a 2-dimensional (2D) representation (e.g., based on two morphological properties as two parameters or dimensions), a 3-dimensional (3D) representation (e.g., based on three morphological properties as three parameters or dimensions), a 4-dimensional (4D) representation, etc. In some examples, one morphological property of a plurality of morphological properties used for blotting the cell morphology map can be represented as a non-axial parameter (e.g., non-x, y, or z axis), such as, distinguishable colors (e.g., heatmap), numbers, letters (e.g., texts of one or more languages), and/or symbols (e.g., a square, oval, triangle, square, etc.). For example, a heatmap can be used as colorimetric scale to represent the classifier prediction percentages for each cell against a cell class, cell type, or cell state.

[0125] The cell morphology map can be generated based on one or more morphological features (e.g., characteristics, profiles, fingerprints, etc.) from the processed image data. Non-limiting examples of one or more morphological properties of a cell, as disclosed herein, that can be extracted from one or more images of the cell may include, but are not limited to (a) shape, curvature, size (e.g., diameter, length, width, circumference), area, volume, texture, thickness,

roundness, etc. of the cell or one or more components of the cell (e.g., cell membrane, nucleus, mitochondria, etc.), (ii) number or positioning of one or more contents (e.g., nucleus, mitochondria, etc.) of the cell within the cell (e.g., center, off-centered, etc.), and (iii) optical characteristics of a region of the image(s) (e.g., unique groups of pixels within the image(s)) that correspond to the cell or a portion thereof (e.g., light emission, transmission, reflectance, absorbance, fluorescence, luminescence, etc.).

[0126] One or more dimension of the cell morphology map can be represented by various approaches (e.g., dimensionality reduction approaches), such as, for example, principal component analysis (PCA), multidimensional scaling (MDS), t-distributed stochastic neighbor embedding (t-SNE), and uniform manifold approximation and projection (UMAP). For example, UMAP can be a machine learning technique for dimension reduction. UMAP can be constructed from a theoretical framework based in Riemannian geometry and algebraic topology. UMAP can be utilized for a practical scalable algorithm that applies to real world data, such as morphological properties of one or more cells.

Training of Deep Learning Model of HFM

[0127] The deep learning model of the human foundation model can be trained using a plurality of cell images from different types of biological samples and thus, be able to detect differences in cell morphology without labeled training data. In some examples, the deep learning model 120 of the human foundation model may be trained using any suitable number of images of cells, for example between about 1 and about 200, about 1 to about 500, between about 1 and about 800, between about 1 and about 1,000, between about 1 and about 2,000, between about 1 and about 5,000, between about 1 and about 8,000, between about 1 and about 10,000, between about 1 and about 20,000, between about 1 and about 50,000, between about 1 and about 80,000, between about 1 and about 100,000, between about 1 and about 200,000, between about 1 and about 500,000, between about 1 and about 800,000, between about 1 and about 1,000,000, between about 1 and about 2,000,000, between about 1 and about 5,000,000, between about 1 and about 8,000,000, or between about 1 and about 10,000,000 images of cells. Any range of the number of cell images as training dataset can be contemplated, for example from about 1 through any number greater than about 10,000,000. As one example, the deep learning model 120 of the human foundation model is trained using a training dataset that includes at least about 10,000 images of cells – e.g., at least about 100,000 images of cells, at least about 1,000,000 images of cells, at least about 5,000,000 images of cells, at least about 10,000,000 images of cells, at least about 100,000,00 images of cells, at least about 1 billion, or more, images of cells. For example, the deep learning model 120 can be

trained using between about 5,000,000 and about 1 billion images of cells. The training set may include, may consist of, or may consist essentially of (and in some examples may consist of), images of cells that are not physically stained and that are not computationally labeled in any manner. As such, in some examples the deep learning model 120 learns to recognize features from the cell images in a self-supervised manner.

[0128] In some examples, the human foundation model may comprise parameters in a range of between about 1 and about 1,000, between about 1 and about 2,000, between about 1 and about 5,000, between about 1 and about 8,000, between about 1 and about 10,000, between about 1 and about 20,000, between about 1 and about 50,000, between about 1 and about 80,000, between about 1 and about 100,000, between about 1 and about 200,000, between about 1 and about 500,000, between about 1 and about 800,000, between about 1 and about 1,000,000, between about 1 and about 2,000,000, between about 1 and about 5,000,000, between about 1 and about 8,000,000, between about 1 and about 10,000,000, between about 1 and about 20,000,000, between about 1 and about 50,000,000, between about 1 and about 80,000,000, between about 1 and about 100,000,000, or between about 1 and about 500,000,000. Any range of the number of parameters can be contemplated, for example from 1 through any number greater than 500,000,000. Some or all of the parameters can be optimized during the training process. For example, a neural network may include millions or billions of floating-point numbers connected by mathematical operations. These numbers in some instances can be called "parameters" or "weights". In some examples, parameters are adjusted ("trained") to transform an image of a cell into a vector (for example classification probabilities, or feature vector, depending on the use-case of the neural network). In some examples, a neural network for computer vision applications such as provided herein may have a number of parameters ranging from 1 million to upwards of 10 billion.

[0129] In some examples, the deep learning model (e.g., backbone model) of the human foundation model, which extracts image features, can be based on a convolutional neural network architecture, a vision transformer architecture, or both. The training process may apply a self-supervised learning approach that learns image features without labels and generate deep learning embeddings (vectors) that are orthogonal to each other and orthogonal to morphometric features. As used herein, embeddings that are "orthogonal" can be perpendicular to another embedding vector or set of embedding vectors. For example, vectors are considered to be orthogonal to each other if they are at right angles in n -dimensional space, where n is the size or number of elements in each vector. In some examples, "orthogonal" embeddings can have a covariance of about 0 and can be perfectly or completely orthogonal (e.g., have exactly a covariance of 0) or be substantially orthogonal with a covariance that is greater than but close to 0. In some examples, "orthogonal"

embeddings include features that are “independent” of one another, meaning, that the presence or absence of one feature does not affect the presence or absence of any of the other feature. For example, a vector is orthogonal if the dot product with another vector is zero.

Examples of Machine Learning Models for Feature Extraction

[0130] In some examples, analysis of imaging data as disclosed herein (e.g., particle imaging data, such as cell imaging data) can be performed using artificial intelligence, such as one or more machine learning algorithms.

[0131] The machine learning model (e.g., a metamodel) can be trained by using a learning model and applying learning algorithms (e.g., machine learning algorithms) on a training dataset (e.g., a dataset comprising unlabeled cell images). In some examples, given a set of training examples/cases, each marked for belonging to a specific class (e.g., specific cell type or class), a training algorithm may build a machine learning model capable of assigning features within images of cells into one category or the other, e.g., to make the model a non-probabilistic machine learning model. In some examples, the machine learning model can be used to create a new category to assign new examples/cases into the new category. In some examples, a machine learning model can be the actual trained model that is generated based on the training model.

[0132] The machine learning algorithm as disclosed herein can be configured to extract one or more morphological features of a cell from the image data of the cell. The machine learning algorithm may form a new data set based on the extracted morphological features, and the new data set need not contain the original image data of the cell. In some examples, replicas of the original images in the image data can be stored in a database disclosed herein, e.g., prior to using any of the new images for training, e.g., to keep the integrity of the images of the image data. In some examples, processed images of the original images in the image data can be stored in a database disclosed herein during or subsequent to the classifier training. In some examples, any of the newly extracted morphological features as disclosed herein can be utilized as new molecular markers for a cell or population of cells of interest to the user. A cell analysis platform as disclosed herein can be operatively coupled to one or more databases comprising non-morphological data of cells processed (e.g., genomics data, transcriptomics data, proteomics data, metabolomics data), a selected population of cells exhibiting the newly extracted morphological feature(s) can be further analyzed by their non-morphological properties to identify proteins or genes of interest that are common in the selected population of cells but not in other cells, thereby determining such proteins or genes of interest to be new molecular markers that can be used to identify such selected population of cells.

[0133] Non-limiting examples of machine learning algorithms for training a machine learning model may include supervised learning, unsupervised learning, semi-supervised learning, reinforcement learning, self-learning (also referred to as self-supervised learning), feature learning, anomaly detection, association rules, etc. In some examples, a machine learning model can be trained by using one or more learning models on such training dataset. Non-limiting examples of learning models may include artificial neural networks (e.g., convolutional neural networks, U-net architecture neural network, etc.), backpropagation, boosting, decision trees, support vector machines, regression analysis, Bayesian networks, genetic algorithms, kernel estimators, conditional random field, random forest, ensembles of machine learning models, minimum complexity machines (MCM), probably approximately correct learning (PACT), etc.

[0134] In some examples, the neural networks are designed by the modification of neural networks such as AlexNet, VGGNet, GoogLeNet, ResNet (residual networks), DenseNet, and Inception networks. In some examples, the enhanced neural networks are designed by modification of ResNet (e.g. ResNet 18, ResNet 34, ResNet 50, ResNet 101, and ResNet 152) or inception networks. In some examples, the modification comprises a series of network surgery operations that are mainly carried out to improve including inference time and/or inference accuracy.

[0135] The neural network can be used together with a Vision Transformer. Vision Transformers and their use in encoding images are described in Dosovitskiy et al., “An image is worth 16x16 words: Transformers for image recognition at scale,” International Conference on Learning Representations (ICLR) (2021) (21 pages available at arxiv.org/abs/2010.11929), the entire contents of which are incorporated by reference herein.

[0136] The machine learning algorithm as disclosed herein may utilize one or more clustering algorithms to determine that objects (e.g., features) in the same cluster can be more similar (in one or more morphological features) to each other than those in other clusters. Non-limiting examples of the clustering algorithms may include, but are not limited to, connectivity models (e.g., hierarchical clustering), centroid models (e.g. K-means algorithm), distribution models (e.g., expectation-maximization algorithm), density models (e.g., density-based spatial clustering of applications with noise (DBSCAN), ordering points to identify the clustering structure (OPTICS)), subspace models (e.g., biclustering), group models, graph-based models (e.g., highly connected subgraphs (HCS) clustering algorithms), single graph models, and neural models (e.g., using unsupervised neural network). The machine learning algorithm may utilize a plurality of models, e.g., in equal weights or in different weights. In some examples, the graph-based models may include graph-based clustering algorithms that use modularity, e.g., such as described in the following references, the entire contents of each of which are incorporated by reference herein:

Blondel et al., “Fast unfolding of communities in large networks,” *Journal of Statistical Mechanics: Theory and Experiment* 2008: P10008 (2008); and Traag et al., “From Louvain to Leiden: guaranteeing well-connected communities,” *Scientific Reports* 9: 5233, 12 pages, (2019).

[0137] In some examples, unsupervised and self-supervised approaches can be used to expedite labeling of image data of cells (extract features from cells). For the example of unsupervised, an embedding for a cell image can be generated. For example, the embedding can be a representation of the image in a space with reduced dimensions than the original image data. Such embeddings can be used to cluster images that are similar to one another. Thus, the labeler can be configured to batch-label the cells and increase the throughput as compared to manually labeling one or more cells.

[0138] In some examples, for the example of self-supervised learning, additional meta information (e.g., additional non-morphological information) about the sample (e.g., what disease is known or associated with the patient who provided the sample) can be used for labeling of image data of cells.

[0139] In some examples, embedding generation may use a neural net trained on predefined cell types. To generate the embeddings described herein, an intermediate layer of the neural net that is trained on predetermined image data (e.g., image data of known cell types and/or states) can be used.

[0140] In some examples, by providing enough diversity in image data or sample data to the trained model, this method may have a benefit of providing an accurate way to cluster future cells.

[0141] In some examples, embedding generation may use neural nets trained for different tasks. To generate the embeddings described herein, an intermediate layer of the neural net that is trained for a different task (e.g., a neural net that is trained on a canonical dataset such as ImageNet). Without wishing to be bound by any particular theory, this can allow the system to focus on features that matter for image classification (e.g., edges and curves) while removing a bias that may otherwise be introduced in labeling the image data.

[0142] In some examples, autoencoders can be used for embedding generation. To generate the embeddings described herein, autoencoders can be used, in which the input and the output can be substantially the same image and the squeeze layer can be used to extract the embeddings. The squeeze layer may force the model to learn a smaller representation of the image, which smaller representation may have sufficient information to recreate the image (e.g., as the output).

[0143] In some examples, for clustering-based labeling of image data or cells, as disclosed herein, an expanding training data set can be used. With the expanding training data set, one or more revisions of labeling (e.g., manual relabeling) can be needed to, for example, avoid the

degradation of model performance due to the accumulated effect of mislabeled images. Such manual relabeling can be intractable on a large scale and ineffective when done on a random subset of the data. Thus, in some examples, to systematically surface images for potential relabeling, similar embedding-based clustering can be used to identify labeled images that may cluster with members of other classes. Such examples are likely to be enriched for incorrect or ambiguous labels, which can be removed (e.g., automatically or manually).

[0144] In some examples, adaptive image augmentation can be used. In order to make the models disclosed herein more robust to artifacts in the image data, (1) one or more images with artifacts can be identified, and (2) such images identified with artifacts can be added to training pipeline (e.g., for training the model). Identifying the image(s) with artifacts may comprise: (1a) while imaging cells, one or more additional sections of the image frame can be cropped, which frame(s) being expected to contain just the background without any cell; (2a) the background image can be checked for any change in one or more characteristics (e.g., optical characteristics, such as brightness); and (3a) flagging/labeling one or more images that have such change in the characteristic(s). Adding the identified images to training pipeline may comprise: (2a) adding the one or more images that have been flagged/labeled as augmentation by first calculating an average feature of the changed characteristic(s) (e.g., the background median color); (2b) creating a delta image by subtracting the average feature from the image data (e.g., subtracting the median for each pixel of the image); and (3c) adding the delta image to the training pipeline.

[0145] In any of the methods or platforms disclosed herein, the model(s) can be validated (e.g., for the ability to demonstrate accurate cell classification performance). Non-limiting examples of validation metrics that can be utilized may include, but are not limited to, threshold metrics (e.g., accuracy, F-measure, Kappa, Macro-Average Accuracy, Mean-Class-Weighted Accuracy, Optimized Precision, Adjusted Geometric Mean, Balanced Accuracy, etc.), the ranking methods and metrics (e.g., receiver operating characteristics (ROC) analysis or “ROC area under the curve (ROC AUC)”), and the probabilistic metrics (e.g., root-mean-squared error). For example, the model(s) can be determined to be balanced or accurate when the ROC AUC is greater than about 0.5 - e.g., greater than about 0.55, greater than about 0.6, greater than about 0.65, greater than about 0.7, greater than about 0.75, greater than about 0.8, greater than about 0.85, greater than about 0.9, greater than about 0.91, greater than about 0.92, greater than about 0.93, greater than about 0.94, greater than about 0.95, greater than about 0.96, greater than about 0.97, greater than about 0.98, greater than about 0.99, or higher.

[0146] As noted further above, the output of the machine learning encoder (model) may include, may consist of, or may consist essentially of, at least one multidimensional vector (which may also

be referred to herein as an embedding). Elements of the vector(s) for a given image may correspond to the values of respective features that the machine learning encoder extracted from that image. Table 1 below describes example machine learning dimensions (for example, deep learning dimensions), which correspond to different features that the machine learning encoder extracts from images. In some examples, the machine learning encoder extracts n ML-based features from each image (where n is a positive integer), and outputs an array of length n , which array can be considered to be an n -dimensional vector. For the illustrative machine learning dimensions listed in Table 1, the output of the deep learning encoder may have the format:

$$[V_1 \quad V_2 \quad V_3 \dots V_n],$$

where the subscripts $1 \dots n$ correspond to the respective deep learning dimension numbers, and wherein the letter V represents the value of the feature in that image that the deep learning encoder calculated. The value of n can be in any suitable range, e.g., can be between about 5 and about 1000 - e.g., between about 10 and about 500, between about 50 and about 100, or range in between any of the aforementioned values. In the nonlimiting example shown in Table 1, n is equal to 64.

TABLE 1. EXAMPLE DEEP LEARNING DERIVED FEATURES GENERATED USING THE HUMAN FOUNDATION MODEL.

Deep learning dimension number	Description
HFMv1:DL001	1 st quantitative description of cell features derived from the HFM model
HFMv1:DL002	2 nd quantitative description of cell features derived from the HFM model
...	...
HFMv1:DL064	64 th quantitative description of cell features derived from the HFM model

[0147] In one example, the ML-based features are not human-interpretable. In one example, because the ML-based features are identified using machine learning, AI, or both, the features are not human-interpretable. For example, the elements of the vector generated by the machine learning encoder may have numeric values, such as **[0.1 4 2.3 ... 10]**, that correspond to the quantitative “amount” of certain features that the machine learning encoder has identified as being present or not in a given image. However, in some examples it may not be possible to identify the meaning of these features, for example as they may correspond to features that are identified by neurons of a convolutional neural network and it is not possible in these examples to reconstruct what those neurons considered to be a feature.

Cell Classification

[0148] In some examples, analysis of imaging data as disclosed herein (e.g., particle imaging

data, such as cell imaging data) can be performed using artificial intelligence, such as one or more machine learning algorithms. In some examples, one or more machine learning models can be used to automatically sort or categorize particles (e.g., cells) in the imaging data into one or more classes (e.g., one or more physical characteristics or morphological features, as used interchangeably herein). In some examples, cell imaging data can be analyzed using the machine learning algorithm(s) to classify (e.g., sort) a cell (e.g., a single cell) in a cell image or video. In some examples, cell imaging data can be analyzed using the machine learning algorithm(s) to determine a focus score of a cell (e.g., a single cell) in a cell image or video. In some examples, cell imaging data can be analyzed using the machine learning algorithm(s) to determine a relative distance between (i) a first plane of cells exhibiting first similar physical characteristic(s) and (ii) a second plane of cells exhibiting second similar physical characteristic(s), which first and second planes denote fluid streams flowing substantially parallel to each other in a channel. In some examples, one or more cell morphology maps as disclosed herein can be used to train one or more machine learning models (e.g., at least about 1, at least about 2, at least about 3, at least about 4, at least about 5, at least about 6, at least about 7, at least about 8, at least about 9, at least about 10, or more machine learning models) as disclosed herein. Each machine learning model can be trained to analyze one or more images of a cell (e.g., to extract one or more morphological features of the cell) and categorize (or classify) the cell into one or more determined class or categories of a cell (e.g., based on a type of state of the cell). In another example, the machine learning model can be trained to create a new category to categorize (or classify) the cell into the new category, e.g., when determining that the cell is morphologically distinct than any pre-existing categories of other cells.

[0149] In some examples, the entire process of cell focusing as disclosed herein (e.g., partitioning of cells into one or more planar currents flowing through the channel) can be accomplished based on *de novo* AI-mediated analysis of each cell (e.g., using analysis of one or more images of each cell using machine learning algorithm). This can be a complete AI or a full AI approach for cell sorting and analysis. In another example, a hybrid approach can be utilized, wherein AI-mediated analysis may analyze cells and one or more heterologous markers that are co-partitioned with the cells (e.g., into the same planar current flowing through the channel), confirm or determine the co-partitioning, after which a more conventional approach (e.g., imaging to detect presence of the heterologous markers, such as fluorescent imaging) can be utilized to sort a subsequent population of cells and the heterologous markers that are co-partitioned into the same planar current.

[0150] The machine learning model (e.g., a metamodel) can be trained by using a learning model and applying learning algorithms (e.g., machine learning algorithms) on a training dataset

(e.g., a dataset comprising examples of specific classes). In some examples, given a set of training examples/cases, each marked for belonging to a specific class (e.g., specific cell type or class), a training algorithm may build a machine learning model capable of assigning new examples/cases (e.g., new datapoints of a cell or a group of cells) into one category or the other, e.g., to make the model a non-probabilistic machine learning model. In some examples, the machine learning model can be capable of creating a new category to assign new examples/cases into the new category. In some examples, a machine learning model can be the actual trained model that is generated based on the training model.

[0151] The machine learning algorithm as disclosed herein can be configured to extract one or more morphological features of a cell from the image data of the cell. The machine learning algorithm may form a new data set based on the extracted morphological features, and the new data set need not contain the original image data of the cell. In some examples, replicas of the original images in the image data can be stored in a database disclosed herein, e.g., prior to using any of the new images for training, e.g., to keep the integrity of the images of the image data. In some examples, processed images of the original images in the image data can be stored in a database disclosed herein during or subsequent to the classifier training. In some examples, any of the newly extracted morphological features as disclosed herein can be utilized as new molecular markers for a cell or population of cells of interest to the user. As cell analysis platform as disclosed herein can be operatively coupled to one or more databases comprising non-morphological data of cells processed (e.g., genomics data, transcriptomics data, proteomics data, metabolomics data), a selected population of cells exhibiting the newly extracted morphological feature(s) can be further analyzed by their non-morphological properties to identify proteins or genes of interest that are common in the selected population of cells but not in other cells, thereby determining such proteins or genes of interest to be new molecular markers that can be used to identify such selected population of cells.

[0152] In some examples, a machine learning model can be trained by applying machine learning algorithms on at least a portion of one or more cell morphology maps as disclosed herein as a training dataset. Non-limiting examples of machine learning algorithms for training a machine learning model may include supervised learning, unsupervised learning, semi-supervised learning, reinforcement learning, self-learning, feature learning, anomaly detection, association rules, etc. In some examples, a machine learning model can be trained by using one or more learning models on such training dataset. Non-limiting examples of learning models may include artificial neural networks (e.g., convolutional neural networks, U-net architecture neural network, etc.), backpropagation, boosting, decision trees, support vector machines, regression analysis, Bayesian

networks, genetic algorithms, kernel estimators, conditional random field, random forest, ensembles of machine learning models, minimum complexity machines (MCM), probably approximately correct learning (PACT), etc.

[0153] In some examples, the neural networks are designed by the modification of neural networks such as AlexNet, VGGNet, GoogLeNet, ResNet (residual networks), DenseNet, and Inception networks. In some examples, the enhanced neural networks are designed by modification of ResNet (e.g. ResNet 18, ResNet 34, ResNet 50, ResNet 101, and ResNet 152) or inception networks. In some examples, the modification comprises a series of network surgery operations that are mainly carried out to improve including inference time and/or inference accuracy.

[0154] The machine learning algorithm as disclosed herein may utilize one or more clustering algorithms to determine that objects (e.g., cells) in the same cluster can be more similar (in one or more morphological features) to each other than those in other clusters. Non-limiting examples of the clustering algorithms may include, but are not limited to, connectivity models (e.g., hierarchical clustering), centroid models (e.g. K-means algorithm), distribution models (e.g., expectation-maximization algorithm), density models (e.g., density-based spatial clustering of applications with noise (DBSCAN), ordering points to identify the clustering structure (OPTICS)), subspace models (e.g., biclustering), group models, graph-based models (e.g., highly connected subgraphs (HCS) clustering algorithms), single graph models, and neural models (e.g., using unsupervised neural network). The machine learning algorithm may utilize a plurality of models, e.g., in equal weights or in different weights.

[0155] In some examples, unsupervised and self-supervised approaches can be used to expedite labeling of image data of cells. For the example of unsupervised, an embedding for a cell image can be generated. For example, the embedding can be a representation of the image in a space with reduced dimensions than the original image data. Such embeddings can be used to cluster images that are similar to one another. Thus, the labeler can be configured to batch-label the cells and increase the throughput as compared to manually labeling one or more cells.

[0156] In some examples, for the example of self-supervised learning, additional meta information (e.g., additional non-morphological information) about the sample (e.g., what disease is known or associated with the patient who provided the sample) can be used for labeling of image data of cells.

[0157] In some examples, embedding generation (e.g., see FIGS. 9-10 and 16-17) may use a neural net trained on predefined cell types. To generate the embeddings described herein, an intermediate layer of the neural net that is trained on predetermined image data (e.g., image data of known cell types and/or states) can be used. By providing enough diversity in image data/sample

data to the trained model, this method may provide an accurate way to cluster future cells.

[0158] In some examples, embedding generation may use neural nets trained for different tasks. To generate the embeddings described herein, an intermediate layer of the neural net that is trained for a different task (e.g., a neural net that is trained on a canonical dataset such as ImageNet). Without wishing to be bound by any particular theory, this may allow to focus on features that matter for image classification (e.g., edges and curves) while removing a bias that may otherwise be introduced in labeling the image data.

[0159] In some examples, autoencoders can be used for embedding generation. To generate the embeddings described herein, autoencoders can be used, in which the input and the output can be substantially the same image and the squeeze layer can be used to extract the embeddings. The squeeze layer may force the model to learn a smaller representation of the image, which smaller representation may have sufficient information to recreate the image (e.g., as the output).

[0160] In some examples, for clustering-based labeling of image data or cells, as disclosed herein, an expanding training data set can be used. With the expanding training data set, one or more revisions of labeling (e.g., manual relabeling) can be needed to, e.g., avoid the degradation of model performance due to the accumulated effect of mislabeled images. Such manual relabeling can be intractable on a large scale and ineffective when done on a random subset of the data. Thus, to systematically surface images for potential relabeling, for example, similar embedding-based clustering can be used to identify labeled images that may cluster with members of other classes. Such examples are likely to be enriched for incorrect or ambiguous labels, which can be removed (e.g., automatically or manually).

[0161] In some examples, adaptive image augmentation can be used. In order to make the models disclosed herein more robust to artifacts in the image data, (1) one or more images with artifacts can be identified, and (2) such images identified with artifacts can be added to training pipeline (e.g., for training the model). Identifying the image(s) with artifacts may comprise: (1a) while imaging cells, one or more additional sections of the image frame can be cropped, which frame(s) being expected to contain just the background without any cell; (2a) the background image can be checked for any change in one or more characteristics (e.g., optical characteristics, such as brightness); and (3a) flagging/labeling one or more images that have such change in the characteristic(s). Adding the identified images to training pipeline may comprise: (2a) adding the one or more images that have been flagged/labeled as augmentation by first calculating an average feature of the changed characteristic(s) (e.g., the background median color); (2b) creating a delta image by subtracting the average feature from the image data (e.g., subtracting the median for each pixel of the image); and (3c) adding the delta image to the training pipeline.

[0162] Because the ML-based features are identified using machine learning, the features may not be human-interpretable. For example, the elements of the vector generated using the machine learning encoder may have numeric values, such as **[0.1 4 2.3 ... 10]**, that correspond to the quantitative “amount” of certain features that the machine learning encoder has identified as being present or not in a given image. However, it may not necessarily be possible to identify the meaning of these features, for example as they may correspond to features that are identified by neurons of a convolutional neural network and it is not possible to reconstruct what those neurons considered to be a feature.

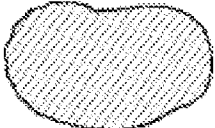
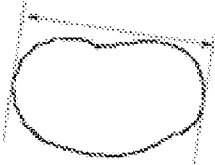
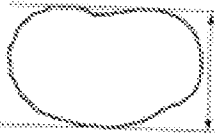
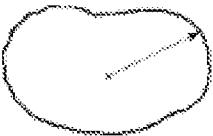
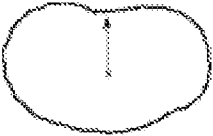
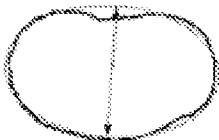
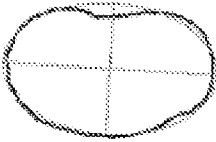
Computer Vision model of HFM

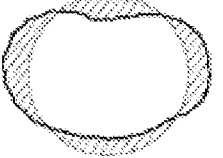
[0163] The computer vision model of the human foundation model may include a set of rules to identify cell morphometric features within an image, and to encode those features into a multidimensional vector. In some examples, the rules can be human defined, and may correspond to features that can be understood by a human.

[0164] As noted further above, the output of the computer vision encoder (model) may include, may consist of, or may consist essentially of, at least one multidimensional vector (which may also be referred to herein as an embedding). Elements of the vector(s) for a given image may correspond to the values of respective features that the computer vision encoder extracted from that image. Because the features are human defined, the features can be human-interpretable. Table 2 below describes example computer vision dimensions (morphometric features), which correspond to different features that the computer vision encoder may extract from images.

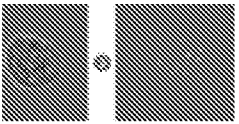
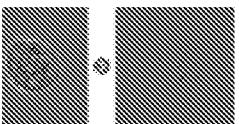
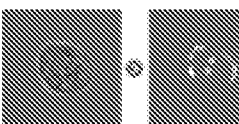
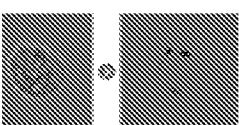
TABLE 2. Example morphometric features generated using the human foundation model. “*” denotes metrics that may also be referred to as blobs or granules.

Category	Morphometric Dimension #	Morphometric Name	Description	Schematic
Position Features	HFMv1:MD001	Centroid X axis (μm)	X axis position of the cell relative to the camera’s field of view	
	HFMv1:MD002	Centroid Y axis (μm)	Y axis position of the cell relative to the camera’s field of view	

Cell Shape Features	HFMv1:MD003	Area (μm^2)	Cell area	
	HFMv1:MD004	Perimeter (μm)	Length of the cell outline	
	HFMv1:MD005	Maximum caliper distance (μm)	Width of widest possible box around the cell	
	HFMv1:MD006	Minimum caliper distance (μm)	Width of narrowest possible box around the cell	
	HFMv1:MD007	Maximum radius (μm)	Largest radius from center of cell-to-cell border	
	HFMv1:MD008	Minimum radius (μm)	Shortest radius from center of cell-to-cell border	
	HFMv1:MD009	Long ellipse axis (μm)	Long axis of best fit ellipse	
	HFMv1:MD010	Short ellipse axis (μm)	Short axis of best fit ellipse	
	HFMv1:MD011	Ellipse elongation (unitless)	Aspect ratio of best fit ellipse Metric: 0 indicates a circle, 1 indicates a line	

	HFMv1:MD012	Ellipse similarity (unitless)	Deviation from an elliptical shape Metric: 0 indicates a perfectly elliptical shape	
	HFMv1:MD013	Roundness (unitless)	Roundness is a measure for circularity or compactness of the shape Metric: Range is 0 to 1, where 1 is a perfect circle	
	HFMv1:MD014	Circle similarity (unitless)	Deviation from a circular shape Metric: 0 indicates a perfectly circular shape	
	HFMv1:MD015	Convex shape (unitless)	Ratio of the area of the convex hull of the cell to the total area of the cell	

Pixel Intensity Features	HFMv1:MD016	Mean pixel intensity (arbitrary units)	Mean pixel grayscale value, refers to how much light a cell absorbs and/or scatters Metric: Range is - 1 to 1	
	HFMv1:MD017	Standard deviation of pixel intensity	Standard deviation of pixel grayscale values gives an indication of the uniformity of pixel intensity within the cell	
	HFMv1:MD018	Pixel intensity 25th percentile	25th percentile of pixel grayscale values	
	HFMv1:MD019	Pixel intensity 75th percentile	75th percentile of pixel grayscale values	
	HFMv1:MD020	Positive fraction	Fraction of pixels with a grayscale value significantly above background	
	HFMv1:MD021	Negative fraction	Fraction of pixels with a grayscale value significantly below background	

Texture Features	HFMv1:MD022	Small set of connected bright pixels*, integral	Sum of pixel intensities in regions identified as being small bright structures Small is described as ~8 pixels but smaller sets could be detected	
	HFMv1:MD023	Small set of connected dark pixels*, integral	Sum of pixel intensities in regions identified as being small dark structures Small is defined as ~8 pixels but smaller sets could be detected	
	HFMv1:MD024	Large set of connected bright pixels*, integral	Sum of pixel intensities in regions identified as being large bright structures Large is defined as ~32 pixels but larger sets could be detected	
	HFMv1:MD025	Large set of connected dark pixels*, integral	Sum of pixel intensities in regions identified as being large dark structures Large is defined as ~32 pixels but larger sets could be detected	

	HFMv1:MD026 - HFMv1:MD027	Image moments (2 moments)	Two values that describe the weighted average or distribution of the image pixel intensities within the cell in a rotation and scale-invariant manner	
	HFMv1:MD028 - HFMv1:MD037	Local binary patterns - center (10 patterns)	The 10 Local Binary Pattern (LBP) – Center features determine the texture inside the cell and describe the appearance near the center of the cell	
	HFMv1:MD038 - HFMv1:MD047	Local binary patterns – periphery (10 patterns)	The 10 Local Binary Pattern (LBP) - Periphery features determine the texture at the periphery of the cell and describe the appearance near the edge of the cell	

Focus Features	HFMv1:MD048	Image sharpness (arbitrary units)	A measure of how sharp or smooth the image is. Typically, an out-of-focus image is less sharp than an in-focus image.	
	HFMv1:MD049	Image focus (μm)	Estimate of the distance of the cell to the focal plane of the microscope	
	HFMv1:MD050	Ring width (arbitrary units)	The imaging modality creates a dark or bright ring to appear around the cell, which is larger and more intense the more out of focus the cell is. This feature estimates the width of the ring.	
	HFMv1:MD051	Ring intensity (arbitrary units)	The imaging modality creates a dark or bright ring to appear around the cell, which is larger and more intense the more out of focus the cell is. This feature estimates the intensity of the ring.	

[0165] From Table 2, it can be understood that morphometric features can be categorized into

different groups. For example, cell morphometric features can be selected from the group consisting of position features, cell shape features, pixel intensity features, texture features, and focus features. In some examples, position features can be selected from the group consisting of: centroid X axis and centroid Y axis, where Table 2 provides respective descriptions for such features. In some examples, cell shape features can be selected from the group consisting of: area, perimeter, maximum caliper distance, minimum caliper distance, maximum radius, minimum radius, long ellipse axis, short ellipse axis, ellipse elongation, ellipse similarity, roundness, circle similarity, and convex shape, where Table 2 provides respective example descriptions for such features. In some examples, pixel intensity features are selected from the group consisting of: mean pixel intensity, standard deviation of pixel intensity, pixel intensity 25th percentile, pixel intensity 75th percentile, positive fraction, and negative fraction, where Table 2 provides respective example descriptions for such features. In some examples, texture features can be selected from the group consisting of: small set of connected bright pixels, integral; small set of connected dark pixels, integral; large set of connected bright pixels, integral; large set of connected dark pixels, integral; image moments; local binary patterns – center; local binary patterns – periphery; image sharpness; image focus; ring width; and ring intensity.

[0166] In some examples, the computer vision encoder extracts m morphometric features from each image (where m is a positive integer), and outputs an array of length m , which array can be considered to be an m -dimensional vector. For the illustrative computer vision dimensions listed in Table 2, the output of the computer vision encoder may have the format:

$$[W_1 \quad W_2 \quad W_3 \quad \dots \quad W_m],$$

where the subscripts $1 \dots m$ correspond to the respective computer vision dimension numbers, and wherein the letter W represents the value of the feature in that image that the computer vision encoder calculated. The value of m can be in any suitable range, e.g., can be between about 5 and about 1000, e.g., between about 10 and about 500, e.g., between about 50 and about 100. In the nonlimiting example shown in Table 2, m is equal to 51.

[0167] Because the morphometric features represent features that are visible by both human and computer vision, the features can be human-interpretable. For example, the elements of the vector generated using the computer vision encoder may have numeric values, such as **[5 0.8 1.4 ... 3.7]**, that correspond to the quantitative “amount” of certain features that the computer vision encoder has identified as being present or not in a given image. The meanings of these features also can be understood by a human. For example, based on Table 2 it can be understood that the value of the first element (e.g., 5) is the centroid X axis in μm of the image (meaning the X axis position of the cell relative to the camera’s field of view), the value of the third element (e.g., 1.4)

is the cell area in μm^2 , and so on.

[0168] The computer vision encoder can be implemented using any suitable combination of hardware and software. As one example, the system component which is implementing the HFM may include a processor and a non-volatile computer-readable medium that includes instructions for causing the processor to respectively process cell images using a computer vision encoder. The computer vision encoder can be configured to quantify the characteristics (e.g., to measure dimensions or intensities) of different features within respective cell images, and to output a vector the dimensions (elements) of which correspond to the measured values of those respective characteristics.

Encoding ML-Based Features and Cell Morphometric Features

[0169] As noted elsewhere herein, the set of ML-based features extracted using the machine learning encoder and the set of cell morphometric features extracted using the computer vision encoder can be used to respectively encode the set of ML-based features and the set of cell morphometric features into a plurality of multi-dimensional vectors that represent morphology of a cell in a cell image. In examples in which the machine learning encoder extracts n ML-based features, and the computer-vision encoder extracts m cell morphological features, the multi-dimensional vectors may have $n+m$ dimensions, where n and m are positive integers. Within each of the multi-dimensional vectors, each dimension of the $n+m$ dimensions can be an element of that multi-dimensional vector, e.g., a numeric value. As one example, continuing with the example provided above, the ML-based features and the cell morphometric features can be concatenated to generate a multi-dimensional vector having the format:

$$[V_1 \quad V_2 \quad V_3 \dots V_n \quad W_1 \quad W_2 \quad W_3 \dots W_m],$$

where, similarly as above, the subscripts $1 \dots n$ correspond to the respective deep learning dimension numbers, the letter V represents the value of the feature in that image that the deep learning encoder calculated, the subscripts $1 \dots m$ correspond to the respective computer vision dimension numbers, and the letter W represents the value of the feature in that image that the computer vision encoder calculated. Note that the value of m can be the same as the value of n , in which example the plurality of multi-dimensional vectors extracted using the machine learning encoder and the computer vision encoder may include a same number of each of the ML-based features and the cell morphological features. In another example, the value of m can be different than the value of n , in which example the plurality of multi-dimensional vectors extracted using

the machine learning encoder and the computer vision encoder may include a different number of each of the ML-based features and the cell morphological features. It will further be appreciated that in some examples, an array of length $(n+m)$, or a vector of length $(n+m)$, can be interpreted as being a plurality of multi-dimensional vectors, each such vector having one or more elements.

[0170] In one example, in a manner such as noted further above, the ML-based features can be orthogonal to one another as explained more particularly in FIGs. 9-10. By this, it is meant that the ML-based features may all be different than one another, and may all be uncorrelated to one another. For example, ML-based feature V_1 can be different than, and uncorrelated to, each of ML-based features $V_2 \dots V_n$. In some other examples, the ML-based features can be orthogonal to the cell morphological features.

System for Cell Morphology Analysis

[0171] Cell morphology can be highly indicative of a cell's phenotype and function, but it is also highly dynamic and complex. Traditional analysis of cell morphology by human eyes has significant limitations. Other methods of assessing and characterizing cell morphology are also limited to imaging or sorting with cell labels.

[0172] Some examples of the present disclosure provides a quantitative, high-dimensional, unbiased platform to assess cell morphology and magnify insights into a cell's phenotype and functions. In some examples, the system as described herein may provide imaging of single cells and label-free sorting in one platform. For example, the system may directly capture high-resolution brightfield images of cells in real time. The system may also enable cell sorting based on their morphology without involving any cell labels. The cells may remain viable and minimally perturbed after the sorting process. In addition, the system may allow collection of sorted cells for downstream analysis, for example, single-cell RNA sequencing.

[0173] The system may comprise, or be compatible with the human foundation model for high-dimensional morphological feature analysis. In some examples, the system may comprise or be compatible with a data suite that may allow users to store, visualize, and analyze images and high-dimensional data. Hence, the system may enable the end-to-end process including cell imaging, morphology analysis, sorting, and classification. In some examples, the system may comprise a microfluidic platform. When cells flow through the microfluidic platform, the system may capture high-resolution brightfield images of each individual cell. The images can be processed by the human foundation model for extracting high-dimensional features corresponding to the cells. The system may sort the cells in different categories, based on the distinct morphological features. The imaging, single-cell morphology analysis, sorting, and classification may occur in real time.

[0174] FIG. 2A illustrates system 100 which includes, and illustrates the interaction between, a microfluidics platform 20 (e.g., corresponding to any microfluidics platform of this disclosure), the human foundation model 60 (e.g., a deep learning model), and a data suite 40, in accordance with some examples of the present disclosure. The system 100 for cell morphology analysis may include microfluidics platform 20, which may include or be compatible with the human foundation model 60 and the data suite 40. Example interactions between the microfluidics platform 20, the human foundation model 60, and the data suite 40 will be described in further detail elsewhere herein, including below in accordance with FIG. 2B. In some aspects, brightfield images of single cells are analyzed in real-time by model 60 to generate quantitative AI embeddings that can include reproducible high-dimensional descriptions of cell morphology. In some aspects, morphologically distinct cell groups can be sorted in real-time by system 100 for downstream analysis, including up to approximately 6 populations per run.

[0175] FIG. 2B illustrates an example workflow from high-throughput imaging to cell characterization, classification and sorting based on cell morphology analysis, in accordance with some examples of the present disclosure. The system for cell morphology profiling as described herein may include a benchtop microfluidics platform that captures high-resolution brightfield images of single cells and sort cells in a label-free manner. The microfluidics platform may comprise or be compatible with the human foundation model and a data suite that may allow users to store, visualize, and analyze images and high-dimensional data. The workflow 200 can be streamlined, starting from preparing and loading cells onto the microfluidics platform (operation 210). In some examples of operation 210, samples from established human cell lines or dissociated tissue biopsies in single cell suspension are loaded onto a microfluidic chip. In some examples, the preparation of samples may comprise dissociation of cells into a single-cell suspension and loading the suspension onto the microfluidics platform. Subsequently, the system may capture images of the cells, and the human foundation model may characterize the cells in real time as they flow through the microfluidic chip (operation 220). In some examples of operation 220, images of single cells are captured and analyzed in real-time by the human foundation model to generate multi-dimensional quantitative morphological profiles (operation 230). The human foundation model may process the images of the cells and generate high-dimensional features reflecting the cell morphology. The images and extracted features can be stored in the data suite (operation 240). The human foundation model also visualizes the cell morphology data by, for example, generating user-defined cell clusters based on cell types (also operation 240). The data suite may also provide in-depth data analysis, including selecting cell populations of interest to sort on the microfluidics platform. The system may recover sorted cells in a plurality of collection wells (operation 250),

which can be used for downstream analyses. In addition, the collected morphology data (referred to as embeddings) can be further analyzed as a unique modality, and users can continuously train customized models for specific applications.

[0176] FIG. 3 illustrates an example system for cell morphology analysis, in accordance with some examples of the present disclosure. The system 300 may comprise a benchtop microfluidics platform 310 that captures high-resolution brightfield images of single cells and sort cells in a label-free manner. System 300 also may include data suite 330, which can be implemented using (and integrated with) microfluidics platform 310, or can be implemented using a separate device. Tables 3 and 4, provided further below, list example parameters, specifications, and components of system 300.

[0177] FIG. 3 schematically illustrates an example method for classifying a cell. The method can comprise processing image data 310 comprising tag-free images/videos of single cells (e.g., image data 310 consisting of tag-free images/videos of single cells). Various clustering analysis models 320 as disclosed herein can be used to process the image data 310 to extract one or more morphological properties of the cells from the image data 310, and generate a cell morphology map 330A based on the extracted one or more morphological properties. For example, the cell morphology map 330A can be generated based on two morphological properties as dimension 1 and dimension 2. The cell morphology map 330A can comprise one or more clusters (e.g., clusters A, B, and C) of datapoints, each datapoint representing an individual cell from the image data 310. The cell morphology map 330A and the clusters A-C therein can be used to train classifier(s) 350. Subsequently, a new image 340 of a new cell can be obtained and processed by the trained classifier(s) 350 to automatically extract and analyze one or more morphological features from the cellular image 340 and plot it as a datapoint on the cell morphology map 330A. Based on its proximity, correlation, or commonality with one or more of the morphologically-distinct clusters A-C on the cell morphology map 330A, the classifier(s) 350 can automatically classify the new cell. The classifier(s) 350 can determine a probability that the cell in the new image data 340 belongs to cluster C (e.g., the likelihood for the cell in the new image data 340 to share one or more commonalities and/or characteristics with cluster C more than with other clusters A/B). For example, the classifier(s) 350 can determine and report that the cell in the new image data 340 has a 95% probability of belonging to cluster C, 1% probability of belonging to cluster B, and 4% probability of belong to cluster A, solely based on analysis of the tag-free image 340 and one or more morphological features of the cell extracted therefrom.

[0178] An image and/or video (e.g., a plurality of images and/or videos) of one or more cells as disclosed herein (e.g., that of image data 310 in FIG. 3) can be captured while the cell(s) is

suspended in a fluid (e.g., an aqueous liquid, such as a buffer) and/or while the cell(s) is moving (e.g., transported across a microfluidic channel). For example, the cell need not be suspended in a gel-like or solid-like medium. The fluid can comprise a liquid that is heterologous to the cell(s)'s natural environment. For example, cells from a subject's blood can be suspended in a fluid that comprises (i) at least a portion of the blood and (ii) a buffer that is heterologous to the blood. The cell(s) may be not immobilized (e.g., embedded in a solid tissue or affixed to a microscope slide, such as a glass slide, for histology) or adhered to a substrate. The cell(s) can be isolated from the natural environment or niche (e.g., a part of the tissue the cell(s) it would be in if not retrieved from a subject by human intervention) when the image and/or video of the cell(s) is captured. For example, the image and/or video need not be from a histological imaging. The cell(s) need not be sliced or sectioned prior to obtaining the image and/or video of the cell, and, as such, the cell(s) may remain substantially intact as a whole during capturing of the image and/or video.

[0179] When the image data is processed, e.g., to extract one or more morphological features of a cell, each cell image can be annotated with the extracted one or more morphological features and/or with information that the cell image belongs to a particular cluster (e.g., a probability).

[0180] The cell morphology map can be a visual (e.g., graphical) representation of one or more clusters of datapoints. The cell morphology map can be a 1-dimensional (1D) representation (e.g., based on one morphological property as one parameter or dimension) or a multi-dimensional representation, such as a 2-dimensional (2D) representation (e.g., based on two morphological properties as two parameters or dimensions), a 3-dimensional (3D) representation (e.g., based on three morphological properties as three parameters or dimensions), a 4-dimensional (4D) representation, etc. In some examples, one morphological property of a plurality of morphological properties used for blotting the cell morphology map can be represented as a non-axial parameter (e.g., non-x, y, or z axis), such as, distinguishable colors (e.g., heatmap), numbers, letters (e.g., texts of one or more languages), and/or symbols (e.g., a square, oval, triangle, square, etc.). For example, a heatmap can be used as colorimetric scale to represent the classifier prediction percentages for each cell against a cell class, cell type, or cell state.

[0181] The cell morphology map can be generated based on one or more morphological features (e.g., characteristics, profiles, fingerprints, etc.) from the processed image data. Non-limiting examples of one or more morphological properties of a cell, as disclosed herein, that can be extracted from one or more images of the cell can include, but are not limited to (i) shape, curvature, size (e.g., diameter, length, width, circumference), area, volume, texture, thickness, roundness, etc. of the cell or one or more components of the cell (e.g., cell membrane, nucleus, mitochondria, etc.), (ii) number or positioning of one or more contents (e.g., nucleus, mitochondria,

etc.) of the cell within the cell (e.g., center, off-centered, etc.), and (iii) optical characteristics of a region of the image(s) (e.g., unique groups of pixels within the image(s)) that correspond to the cell or a portion thereof (e.g., light emission, transmission, reflectance, absorbance, fluorescence, luminescence, etc.).

[0182] Non-limiting examples of clustering as disclosed herein can be hard clustering (e.g., determining whether a cell belongs to a cluster or not), soft clustering (e.g., determining a likelihood that a cell belongs to each cluster to a certain degree), strict partitioning clustering (e.g., determining whether each cell belongs to exactly one cluster), strict partitioning clustering with outliers (e.g., determining whether a cell can also belong to no cluster), overlapping clustering (e.g., determining whether a cell can belong to more than one cluster), hierarchical clustering (e.g., determining whether cells that belong to a child cluster can also belong to a parent cluster), and subspace clustering (e.g., determining whether clusters are not expected to overlap).

[0183] Cell clustering and/or generation of the cell morphology map, as disclosed herein, can be based on a single morphological property of the cells. In another example, cell clustering and/or generation the cell morphology map can be based on a plurality of different morphological properties of the cells. In some examples, the plurality of different morphological properties of the cells can have the same weight or different weights. A weight can be a value indicative of the importance or influence of each morphological property relative to one another in training the classifier or using the classifier to (i) generate one or more cell clusters, (ii) generate the cell morphology map, or (iii) analyze a new cellular image to classify the cellular image as disclosed herein. For example, cell clustering can be performed by having 50% weight on cell shape, 40% weight on cell area, and 10% weight on texture (e.g., roughness) of the cell membrane. In some examples, the classifier as disclosed herein can be configured to adjust the weights of the plurality of different morphological properties of the cells during analysis of new cellular image data, thereby to yield a most optimal cell clustering and cell morphology map. The plurality of different morphological properties with different weights can be utilized during the same analysis operation for cell clustering and/or generation of the cell morphology map.

[0184] The plurality of different morphological properties can be analyzed hierarchically. In some examples, a first morphological property can be used as a parameter to analyze image data of a plurality of cells to generate an initial set of clusters. Subsequently, a second and different morphological property can be used as a second parameter to (i) modify the initial set of clusters (e.g., optimize arrangement among the initial set of clusters, re-group some clusters of the initial set of clusters, etc.) and/or (ii) generate a plurality of sub-clusters within a cluster of the initial set of clusters. In some examples, a first morphological property can be used as a parameter to analyze

image data of a plurality of cells to generate an initial set of clusters, to generate a 1D cell morphology map. Subsequently, a second morphological property can be used as a parameter to further analyze the clusters of the 1D cell morphology map, to modify the clusters and generate a 2D cell morphology map (e.g., a first axis parameter based on the first morphological property and a second axis parameter based on the second morphological property).

[0185] In some examples of the hierarchical clustering as disclosed herein, an initial set of clusters can be generated based on an initial morphological feature that is extracted from the image data, and one or more clusters of the initial set of clusters can comprise a plurality of sub-clusters based on second morphological features or sub-features of the initial morphological feature. For example, the initial morphological feature can be cell type, such as stem cells (or not), and the sub-features can be different types of stem cells (e.g., embryonic stem cells, induced pluripotent stem cells, mesenchymal stem cells, muscle stem cells, etc.). In another example, the initial can be cancer cells (or not), and the sub-feature can be different types of cancer cells (e.g., sarcoma cells, sarcoma cells, leukemia cells, lymphoma cells, multiple myeloma cells, melanoma cells, etc.). In a different example, the initial can be cancer cells (or not), and the sub-feature can be different stages of the cancer cell (e.g., quiescent, proliferative, apoptotic, etc.).

[0186] Each datapoint can represent an individual cell or a collection of a plurality of cells (e.g., at least about 2 - e.g., at least about 3, at least about 4, at least about 5, at least about 6, at least about 7, at least about 8, at least about 9, or at least about 10 cells Or more). Each datapoint can represent an individual image (e.g., of a single cell or a plurality of cells) or a collection of a plurality of images (e.g., at least about 2, at least about 3, at least about 4, at least about 5, at least about 6, at least about 7, at least about 8, at least about 9, or at least about 10 images of the same single cell or different cells or more).

[0187] The cell morphology map can comprise at least about 1, or at least about 2, or at least about 3, or at least about 4, or at least about 5, or at least about 6, or at least about 7, or at least about 8, or at least about 9, or at least about 10, or at least about 15, or at least about 20, or at least about 30, or at least about 40, or at least about 50, or at least about 60, or at least about 70, or at least about 80, or at least about 90, or at least about 100, or at least about 150, or at least about 200, or at least about 300, or at least about 400, or at least about 500 clusters, or more.

[0188] Each cluster as disclosed herein can comprise a plurality of sub-clusters, e.g., at least about 2, or at least about 3, or at least about 4, or at least about 5, or at least about 6, or at least about 7, or at least about 8, or at least about 9, or at least about 10, or at least about 15, or at least about 20, or at least about 30, or at least about 40, or at least about 50, or at least about 60, or at least about 70, or at least about 80, or at least about 90, or at least about 100, or at least about 150,

or at least about 200, or at least about 300, or at least about 400, or at least about 500 sub-clusters.

[0189] A cluster (or sub-cluster) can comprise datapoints representing cells of the same type/state. In another example, a cluster (or sub-cluster) can comprise datapoints representing cells of different types/states.

[0190] A cluster (or sub-cluster) can comprise at least about 1, at least about 2, at least about 3, at least about 4, at least about 5, at least about 6, at least about 7, at least about 8, at least about 9, at least about 10, at least about 15, at least about 20, at least about 30, at least about 40, at least about 50, at least about 60, at least about 70, at least about 80, at least about 90, at least about 100, at least about 150, at least about 200, at least about 300, at least about 400, at least about 500, at least about 1,000, at least about 2,000, at least about 3,000, at least about 4,000, at least about 5,000, at least about 10000, at least about 50,000, or at least about 100,000 datapoints.

[0191] Two or more clusters may overlap in a cell morphology map. In another example, no clusters may not overlap in a cell morphology map. In some examples, an allowable degree of overlapping between two or more clusters can be adjustable (e.g., manually or automatically by a machine learning algorithm) depending on the quality, condition, or size of data in the image data being processed.

[0192] A cluster (or sub-cluster) as disclosed herein can be represented with a boundary (e.g., a solid line or a dashed line). In another example, a cluster or sub-cluster need not be represented with a boundary, and can be distinguishable from other cluster(s) sub-cluster(s) based on their proximity to one another.

[0193] A cluster (or sub-cluster) or a data comprising information about the cluster can be annotated based on one or more annotation schema (e.g., predefined annotation schema). Such annotation can be manual (e.g., by a user of the method or system disclosed herein) or automatically (e.g., by any of the machine learning algorithms disclosed herein). The annotation of the clustering can be related the one or more morphological properties of the cells that have been analyzed (e.g., cell shape, cell area, optical characteristic(s), etc.) to generate the cluster or assign one or more datapoints to the cluster. In another example, the annotation of the clustering can be related to information that has not been used or analyzed to generate the cluster or assign one or more datapoints to the cluster (e.g., genomics, transcriptomics, or proteomics, etc.). In such example, the annotation can be utilized to add additional “layers” of information to each cluster.

[0194] In some examples, an interactive annotation tool can be provided that permits one or more users to modify any process of the method described herein. For example, the interactive annotation tool can allow a user to curate, verify, edit, and/or annotate the morphologically-distinct clusters. In another example, the interactive annotation tool can process the image data, extract one

or more morphological features from the image data, and allow the user to select one or more of the extracted morphological features to be used as a basis to generate the clusters and/or the cell morphology map. After the generation of the clusters and/or the cell morphology map, the interactive annotation tool can allow the user to annotate each cluster and/or the cell morphology map using (i) a predefined annotation schema or (ii) a new, user-defined annotation schema. In another example, the interactive annotation tool can allow user to assign different weights to different morphological features for the clustering and/or map plotting. In another example, the interactive annotation tool can allow user to select with imaging data (or which cells) to be used and/or which imaging data (or which cells, cell clumps, artifacts, or debris) to be discarded, for the clustering and/or map plotting. A user can manually identify incorrectly clustered cells, or the machine learning algorithm can provide probability or correlation value of cells within each cluster and identify any outlier (e.g., a datapoint that would change the outcome of the probability/correlation value of the cluster(s) by a certain percentage value). Thus, the user can choose to move the outliers using the interactive annotation tool to further tune the cell morphology map, e.g., to yield a “higher resolution” map.

[0195] One or more cell morphology maps as disclosed herein can be used to train one or more classifiers (e.g., at least about 1, at least about 2, at least about 3, at least about 4, at least about 5, at least about 6, at least about 7, at least about 8, at least about 9, at least about 10, or more classifiers) as disclosed herein. Each classifier can be trained to analyze one or more images of a cell (e.g., to extract one or more morphological features of the cell) and categorize (or classify) the cell into one or more determined class or categories of a cell (e.g., based on a type of state of the cell). In another example, the classifier can be trained to create a new category to categorize (or classify) the cell into the new category, e.g., when determining that the cell is morphologically distinct than any pre-existing categories of other cells.

[0196] The machine learning algorithm as disclosed herein can be configured to extract one or more morphological feature of a cell from the image data of the cell. The machine learning algorithm can form a new data set based on the extracted morphological features, and the new data set need not contain the original image data of the cell. In some examples, replicas of the original images in the image data can be stored in a database disclosed herein, e.g., prior to using any of the new images for training, e.g., to keep the integrity of the images of the image data. In some examples, processed images of the original images in the image data can be stored in a database disclosed herein during or subsequent to the classifier training. In some examples, any of the newly extracted morphological features as disclosed herein can be utilized as new molecular markers for a cell or population of cells of interest to the user. As cell analysis platform as disclosed herein can

be operatively coupled to one or more databases comprising non-morphological data of cells processed (e.g., genomics data, transcriptomics data, proteomics data, metabolomics data), a selected population of cells exhibiting the newly extracted morphological feature(s) can be further analyzed by their non-morphological properties to identify proteins or genes of interest that are common in the selected population of cells but not in other cells, thereby determining such proteins or genes of interest to be new molecular markers that can be used to identify such selected population of cells.

[0197] In some examples, a classifier can be trained by applying machine learning algorithms on at least a portion of one or more cell morphology maps as disclosed herein as a training dataset. Non-limiting examples of machine learning algorithms for training a classifier can include supervised learning, unsupervised learning, semi-supervised learning, reinforcement learning, self-learning, feature learning, anomaly detection, association rules, etc. In some examples, a classifier can be trained by using one or more learning models on such training dataset. Non-limiting examples of learning models can include artificial neural networks (e.g., convolutional neural networks, U-net architecture neural network, etc.), backpropagation, boosting, decision trees, support vector machines, regression analysis, Bayesian networks, genetic algorithms, kernel estimators, conditional random field, random forest, ensembles of classifiers, minimum complexity machines (MCM), probably approximately correct learning (PACT), etc.

[0198] In some examples, the neural networks are designed by the modification of neural networks such as AlexNet, VGGNet, GoogLeNet, ResNet (residual networks), DenseNet, and Inception networks. In some examples, the enhanced neural networks are designed by modification of ResNet (e.g. ResNet 18, ResNet 34, ResNet 50, ResNet 101, and ResNet 152) or inception networks. In some examples, the modification comprises a series of network surgery operations that are mainly carried out to improve including inference time and/or inference accuracy.

[0199] The machine learning algorithm as disclosed herein can utilize one or more clustering algorithms to determine that objects in the same cluster can be more similar (in one or more morphological features) to each other than those in other clusters. Non-limiting examples of the clustering algorithms can include, but are not limited to, connectivity models (e.g., hierarchical clustering), centroid models (e.g. K-means algorithm), distribution models (e.g., expectation-maximization algorithm), density models (e.g., density-based spatial clustering of applications with noise (DBSCAN), ordering points to identify the clustering structure (OPTICS)), subspace models (e.g., biclustering), group models, graph-based models (e.g., highly connected subgraphs (HCS) clustering algorithms), single graph models, and neural models (e.g., using unsupervised neural network). The machine learning algorithm can utilize a plurality of models, e.g., in equal

weights or in different weights.

[0200] In some examples, unsupervised and self-supervised approaches can be used to expedite labeling of image data of cells. For the example of unsupervised, an embedding for a cell image can be generated. For example, the embedding can be a representation of the image in a space with reduced dimensions than the original image data. Such embeddings can be used to cluster images that are similar to one another. Thus, the labeler can be configured to batch-label the cells and increase the throughput as compared to manually labeling one or more cells.

[0201] In some examples, for the example of self-supervised learning, additional meta information (e.g., additional non-morphological information) about the sample (e.g., what disease is known or associated with the patient who provided the sample) can be used for labeling of image data of cells.

[0202] In some examples, embedding generation can use a neural net trained on predefined cell types. To generate the embeddings described herein, an intermediate layer of the neural net that is trained on predetermined image data (e.g., image data of known cell types and/or states) can be used. By providing enough diversity in image data/sample data to the trained model/classifier, this method can provide an accurate way to cluster future cells.

[0203] In some examples, embedding generation can use neural nets trained for different tasks. To generate the embeddings described herein, an intermediate layer of the neural net that is trained for a different task (e.g., a neural net that is trained on a canonical dataset such as ImageNet). Without wishing to be bound by any particular theory, this can allow to focus on features that matter for image classification (e.g., edges and curves) while removing a bias that may otherwise be introduced in labeling the image data.

[0204] In some examples, autoencoders can be used for embedding generation. To generate the embeddings described herein, autoencoders can be used, in which the input and the output can be substantially the same image and the squeeze layer can be used to extract the embeddings. The squeeze layer can force the model to learn a smaller representation of the image, which smaller representation may have sufficient information to recreate the image (e.g., as the output).

[0205] In some examples, for clustering-based labeling of image data or cells, as disclosed herein, an expanding training data set can be used. With the expanding training data set, one or more revisions of labeling (e.g., manual relabeling) can be needed to, e.g., avoid the degradation of model performance due to the accumulated effect of mislabeled images. Such manual relabeling can be intractable on a large scale and ineffective when done on a random subset of the data. Thus, to systematically surface images for potential relabeling, for example, similar embedding-based clustering can be used to identify labeled images that may cluster with members of other classes.

Such examples are likely to be enriched for incorrect or ambiguous labels, which can be removed (e.g., automatically or manually).

[0206] In some examples, adaptive image augmentation can be used. In order to make the models and classifiers disclosed herein more robust to artifacts in the image data, (1) one or more images with artifacts can be identified, and (2) such images identified with artifacts can be added to training pipeline (e.g., for training the model/classifier). Identifying the image(s) with artifacts can comprise: (1a) while imaging cells, one or more additional sections of the image frame can be cropped, which frame(s) being expected to contain just the background without any cell; (2a) the background image can be checked for any change in one or more characteristics (e.g., optical characteristics, such as brightness); and (3a) flagging/labeling one or more images that have such change in the characteristic(s). Adding the identified images to training pipeline can comprise: (2a) adding the one or more images that have been flagged/labeled as augmentation by first calculating an average feature of the changed characteristic(s) (e.g., the background median color); (2b) creating a delta image by subtracting the average feature from the image data (e.g., subtracting the median for each pixel of the image); and (3c) adding the delta image to the training pipeline.

[0207] One or more dimension of the cell morphology map can be represented by various approaches (e.g., dimensionality reduction approaches), such as, for example, principal component analysis (PCA), multidimensional scaling (MDS), t-distributed stochastic neighbor embedding (t-SNE), and uniform manifold approximation and projection (UMAP). For example, UMAP can be a machine learning technique for dimension reduction. UMAP can be constructed from a theoretical framework based in Riemannian geometry and algebraic topology. UMAP can be utilized for a practical scalable algorithm that applies to real world data, such as morphological properties of one or more cells.

[0208] The cell morphology map as disclosed herein can comprise an ontology of the one or more morphological features. The ontology can be an alternative medium to represent a relationship among various datapoints (e.g., each representing a cell) analyzed from an image data. For example, an ontology can be a data structure of information, in which nodes can be linked by edges. An edge can be used to define a relationship between two nodes. For example, a cell morphology map can comprise a cluster comprising sub-clusters, and the relationship between the cluster and the sub-clusters can be represented in an nodes/edges ontology (e.g., an edge can be used to describe the relationship as a subclass of, genus of, part of, stem cell of, differentiated from, progeny of, diseased state of, targets, recruits, interacts with, same tissue, different tissue, etc.).

[0209] In some examples, one-to-one morphology to genomics mapping can be utilized. An image of a single cell or images of multiple “similar looking” cells can be mapped to its/their

molecular profile(s) (e.g., genomics, proteomics, transcriptomics, etc.). In some examples, classifier-based barcoding can be performed. Each sorting event (e.g., positive classifier) can push the sorted cell(s) into an individual well or droplet with a unique barcode (e.g., nucleic acid or small molecule barcode). The exact barcode(s) used for that individual classifier positive event can be recorded and tracked. Following, the cells can be lysed and molecularly analyzed together with the barcode(s). The result of the molecular analysis can then be mapped (e.g., one-to-one) to the image(s) of the individual (or ensemble of) sorted cell(s) captured while the cell(s) are flowing in the flow channel. In some examples, class-based sorting can be utilized. Cells that are classified in the same class based at least on their morphological features can be sorted into a single well or droplet with a pre-determined barcoded material, and the cells can be lysed, molecularly analyzed, then any molecular information can be used for the one-to-one mapping as disclosed herein.

[0210] FIG. 4 schematically illustrates different ways of representing analysis data of image data of cells. Tag-free image data 410 of cells (e.g., circular cells and square cells) having different nuclei (e.g., small nucleus and large nucleus) can be analyzed by any of the methods disclosed herein (e.g., based on extraction of one or more morphological features). For example, any of the classifier(s) disclosed herein can be used to analyze and plot the image data 410 into a cell morphology map 420, comprising four distinguishable clusters: cluster A (circular cell, small nucleus), cluster B (circular cell, large nucleus), cluster C (square cell, small nucleus), and cluster D (square cell, large nucleus). The classifier(s) can also represent the analysis in a cell morphological ontology 430, in which a top node (“cell shape”) can be connected to two sub-nodes (“circular cell” and “rectangular cell”) using an edge (“is a subclass of”) to define the relationship between the nodes. Each sub-node can also be connected to its own sub-nodes (“small nucleus” and “large nucleus”) using an edge (“is a part of”) to define their relationships. The sub-nodes (e.g., “small nucleus” and “large nucleus”) can also be connected using one or more edges (“are similar”) to further define their relationship.

[0211] The cell morphology map or cell morphological ontology as disclosed herein can be further annotated with one or more non-morphological data of each cell. As shown in FIG. 3, the ontology 430 from FIG. 4 can be further annotated with information about the cells that may not be extractable from the image data used to classify the cells (e.g., molecular profiles obtained using molecular barcodes, as disclosed herein). Non-limiting examples of such non-morphological data can be from additional treatment and/or analysis, including, but not limited to, cell culture (e.g., proliferation, differentiation, etc.), cell permeabilization and fixation, cell staining by a probe, mass cytometry, multiplexed ion beam imaging (MIBI), confocal imaging, nucleic acid (e.g., DNA, RNA) or protein extraction, polymerase chain reaction (PCR), target nucleic acid enrichment,

sequencing, sequence mapping, etc.

[0212] Examples of the probe used for cell staining (or tagging) may include, but are not limited to, a fluorescent probe (e.g., for staining chromosomes such as X, Y, 13, 18 and 21 in fetal cells), a chromogenic probe, a direct immunoagent (e.g. labeled primary antibody), an indirect immunoagent (e.g., unlabeled primary antibody coupled to a secondary enzyme), a quantum dot, a fluorescent nucleic acid stain (such as DAPI, Ethidium bromide, Sybr green, Sybr gold, Sybr blue, Ribogreen, Picogreen, YoPro-1, YoPro-2 YoPro-3, YOYO, Oligreen acridine orange, thiazole orange, propidium iodine, or Hoeste), another probe that emits a photon, or a radioactive probe.

[0213] In some examples, the instrument(s) for the additional analysis may comprise a computer executable logic that performs karyotyping, in situ hybridization (ISH) (e.g., fluorescence in situ hybridization (FISH), chromogenic in situ hybridization (CISH), nanogold in situ hybridization (NISH)), restriction fragment length polymorphism (RFLP) analysis, polymerase chain reaction (PCR) techniques, flow cytometry, electron microscopy, quantum dot analysis, or detects single nucleotide polymorphisms (SNPs) or levels of RNA.

[0214] Analysis of the image data (e.g., extracting one or more morphological features from the image data, determining clustering and/or cell morphology map based on the image data, etc.) can be performed (e.g., automatically) within less than about 1 hour- e.g., less than about 50 minutes, or less than about 40 minutes, or less than about 30 minutes, or less than about 25 minutes, or less than about 20 minutes, or less than about 15 minutes, or less than about 10 minutes, or less than about 9 minutes, or less than about 8 minutes, or less than about 7 minutes, or less than about 6 minutes, or less than about 5 minutes, or less than about 4 minutes, or less than about 3 minutes, or less than about 2 minutes, or less than about 1 minute, or less than about 50 seconds, or less than about 40 seconds, or less than about 30 seconds, or less than about 20 seconds, or less than about 10 seconds, or less than about 5 seconds, about 1 second, or less. In some examples, such a analysis can be performed in real-time.

[0215] One or more morphological features utilized for generating the clusters or the cell morphology map, as disclosed herein, can be selected automatically (e.g., by one or more machine learning algorithms) or, alternatively, selected manually by a user using a user interface (e.g., graphical user interface (GUI)). The GUI can show visualization of, for example, (i) the one or more morphological parameters extracted from the image data (e.g., represented as images, words, symbols, predefined codes, etc.), (ii) the cell morphology map comprising one or more clusters, or (iii) the cell morphological ontology. The user can select, using the GUI, which morphological parameter(s) to be used to generate the clusters and the cell morphological map prior to actual

generation of the clusters and the cell morphological map. The user can, upon seeing or receiving a report about the generated clusters and the cell morphological map, retroactively modify the types of morphological parameter(s) to use, thereby to (i) modify the clustering or the cell morphological mapping and/or (ii) create new cluster(s) or new cell morphological map(s). In some examples, the user can select one or more regions to be excluded or included for further analysis or further processing of the cells (e.g., sorting in the future or in real-time). For example, a microfluidic system as disclosed herein can be utilized to capture image(s) of each cell from a population of cells, and any of the methods disclosed herein can be utilized to analyze such image data to generate a cell morphology map comprising clusters representing the population of cells. The user can select one or more clusters or sub-clusters to be sorted, and the input can be provided to the microfluidic system to sort at least a portion of the cells into one or more sub-channels of the microfluidic system (e.g., in real-time) accordingly. In another example, the user can select one or more clusters or sub-clusters to be excluded during sorting (e.g., to get rid of artifacts, debris, or dead cells), and the input can be provided to the microfluidic system to sort at least a portion of the cells into one or more sub-channels of the microfluidic system (e.g., in real-time) accordingly without such artifacts, debris, or dead cells.

[0216] The cell morphology map or cell morphological ontology as disclosed herein can be further annotated with one or more non-morphological data of each cell. As shown in FIG. 5, the ontology 430 from FIG. 4 can be further annotated with information about the cells that may not be extractable from the image data used to classify the cells (e.g., molecular profiles obtained using molecular barcodes, as disclosed herein). Non-limiting examples of such non-morphological data can be from additional treatment and/or analysis, including, but not limited to, cell culture (e.g., proliferation, differentiation, etc.), cell permeabilization and fixation, cell staining by a probe, mass cytometry, multiplexed ion beam imaging (MIBI), confocal imaging, nucleic acid (e.g., DNA, RNA) or protein extraction, polymerase chain reaction (PCR), target nucleic acid enrichment, sequencing, sequence mapping, etc.

[0217] Examples of the probe used for cell staining (or tagging) may include, but are not limited to, a fluorescent probe (e.g., for staining chromosomes such as X, Y, 13, 18 and 21 in fetal cells), a chromogenic probe, a direct immunoagent (e.g. labeled primary antibody), an indirect immunoagent (e.g., unlabeled primary antibody coupled to a secondary enzyme), a quantum dot, a fluorescent nucleic acid stain (such as DAPI, Ethidium bromide, Sybr green, Sybr gold, Sybr blue, Ribogreen, Picogreen, YoPro-1, YoPro-2 YoPro-3, YOYo, Oligreen acridine orange, thiazole orange, propidium iodine, or Hoeste), another probe that emits a photon, or a radioactive probe.

[0218] In some examples, the instrument(s) for the additional analysis may comprise a computer executable logic that performs karyotyping, in situ hybridization (ISH) (e.g., florescence in situ hybridization (FISH), chromogenic in situ hybridization (CISH), nanogold in situ hybridization (NISH)), restriction fragment length polymorphism (RFLP) analysis, polymerase chain reaction (PCR) techniques, flow cytometry, electron microscopy, quantum dot analysis, or detects single nucleotide polymorphisms (SNPs) or levels of RNA.

[0219] Analysis of the image data (e.g., extracting one or more morphological features from the image data, determining clustering and/or cell morphology map based on the image data, etc.) can be performed (e.g., automatically) within less than about 1 hour, about 50 minutes, about 40 minutes, about 30 minutes, about 25 minutes, about 20 minutes, about 15 minutes, about 10 minutes, about 9 minutes, about 8 minutes, about 7 minutes, about 6 minutes, about 5 minutes, about 4 minutes, about 3 minutes, about 2 minutes, about 1 minute, about 50 seconds, about 40 seconds, about 30 seconds, 20 seconds, about 10 seconds, about 5 seconds, about 1 second, or less. In some examples, such analysis can be performed in real-time.

[0220] One or more morphological features utilized for generating the clusters or the cell morphology map, as disclosed herein, can be selected automatically (e.g., by one or more machine learning algorithms) or, alternatively, selected manually by a user using a user interface (e.g., graphical user interface (GUI)). The GUI can show visualization of, for example, (i) the one or more morphological parameters extracted from the image data (e.g., represented as images, words, symbols, predefined codes, etc.), (ii) the cell morphology map comprising one or more clusters, or (iii) the cell morphological ontology. The user can select, using the GUI, which morphological parameter(s) to be used to generate the clusters and the cell morphological map prior to actual generation of the clusters and the cell morphological map. The user can, upon seeing or receiving a report about the generated clusters and the cell morphological map, retroactively modify the types of morphological parameter(s) to use, thereby to (i) modify the clustering or the cell morphological mapping and/or (ii) create new cluster(s) or new cell morphological map(s). In some examples, the user can select one or more regions to be excluded or included for further analysis or further processing of the cells (e.g., sorting in the future or in real-time). For example, a microfluidic system as disclosed herein can be utilized to capture image(s) of each cell from a population of cells, and any of the methods disclosed herein can be utilized to analyze such image data to generate a cell morphology map comprising clusters representing the population of cells. The user can select one or more clusters or sub-clusters to be sorted, and the input can be provided to the microfluidic system to sort at least a portion of the cells into one or more sub-channels of the microfluidic system (e.g., in real-time) accordingly. In another example, the user can select one or more clusters

or sub-clusters to be excluded during sorting (e.g., to get rid of artifacts, debris, or dead cells), and the input can be provided to the microfluidic system to sort at least a portion of the cells into one or more sub channels of the microfluidic system (e.g., in real-time) accordingly without such artifacts, debris, or dead cells.

[0221] FIG. 6 schematically illustrates a method for a user to interact (e.g., using GUI) with any one of the methods disclosed herein. Image data 610 of a plurality of cells can be processed, using any one of the methods disclosed herein, to generate a cell morphology map 620A that represents the plurality of cells as datapoints in different clusters A, B, C, and D. The cell morphology map 620A can be displayed to the user using the GUI 630. The user can select each cluster or a datapoint within each cluster to visualize one or more images 650a, b, c, or d of the cells classified into the cluster. Upon visualization of the images, the user can draw a box 640 (e.g., using any user-defined shape and/or size) around one or more datapoints or around a cluster. For example, the user can draw a box 640 around a cluster of “debris” datapoints, to, e.g., remove the selected cluster and generate a new cell morphology map 620B. The user input can be used to update cell classifying algorithms, mapping algorithms, cell flowing mechanism (e.g., velocity of cells, positioning of the cells within a flow channel, adjusting imaging focal length/plane of one or more sensors/cameras of an imaging module (also referred to as an imaging device herein) that captures one or more images/videos of cells flowing through the cartridge, etc.), cell sorting mechanisms in the flow channel, cell sorting instructions in the flow channel, etc. For example, upon the user’s selection, the classifier can be trained to identify one or more common morphological features within the selected datapoints (e.g., features that distinguish the selected datapoints from the unselected data). Features of the selected group can be used to further identify other cells from other samples having similar feature(s) for further analysis or discard cells having similar feature(s), e.g., for cell sorting.

[0222] The present disclosure also describes a cell analysis platform, e.g., for analyzing or classifying a cell. The cell analysis platform can be a product of any one of the methods disclosed herein. In another example, or in addition to, the cell analysis platform can be used as a basis to execute any one of the methods disclosed herein. For example, the cell analysis platform can be used to process image data comprising tag-free images of single cells to generate a new cell morphology map of various cell clusters. In another example, the cell analysis platform can be used to process image data comprising tag-free images of single cells to compare the cell to pre-determined (e.g., pre-analyzed) images of known cells or cell morphology map(s), such that the single cells from the image data can be classified, e.g., for cell sorting. FIG. 7 illustrates an example cell analysis platform (e.g., machine learning/artificial intelligence platform) for analyzing image data of one or more cells. The cell analysis platform 700 can comprise a cell morphology atlas

(CMA) 705. The CMA 705 can comprise a database 710 having a plurality of annotated single cell images that are grouped into morphologically-distinct clusters (e.g., represented a texts, as cell morphology map(s), or cell morphological ontology(ies)) corresponding to a plurality of classifications (e.g., predefined cell classes). The CMA 705 can comprise a modeling unit comprising one or more models (e.g., modeling library 720 comprising, such as, one or more machine learning algorithms disclosed herein) that are trained and validated using datasets from the CMA 705, to process image data comprising images/videos of one or more cells to identify different cell types and/or states based at least on morphological features. The CMA 705 can comprise an analysis module 730 comprising one or more classifiers as disclosed herein. The classifier(s) can uses one or more of the models from the modeling library 720 to, e.g., (1) classify one or more images taken from a sample, (2) assess a quality or state of the sample based on the one or more images, (3) map one or more datapoints representing such one or more images onto a cell morphology map (or cell morphological ontology) using a mapping module 740. The CMA 705 can be operatively coupled to one or more additional database 770 to receive the image data comprising the images/videos of one or more cells. For example, the image data from the database 770 can be obtained from an imaging module 792 of a cartridge 790, which can also be operatively coupled to the CMA 705. The cartridge can direct flow of a sample comprising or suspected of comprising a target cell, and capture one or more images of contents (e.g., cells) within the sample by the imaging module 792. Any image data obtained by the imaging module 792 can be transmitted directly to the CMA 705 and/or to the new image database 770. In another example, the CMA 705 can be operatively coupled to one or more additional databases 780 comprising non-morphological data of any of the cells (e.g., genomics, transcriptomics, or proteomics, etc.), e.g., to further annotate any of the datapoint, cluster, map, ontology, images, as disclosed herein. The CMA 705 can be operatively coupled to a user device 750 (e.g., a computer or a mobile device comprising a display) comprising a GUI 760 for the user to receive information from and/or to provide input (e.g., instructions to modify or assist any portion of the method disclosed herein). Any classification made by the CMA and/or the user can be provided as an input to the sorting module 794 of the cartridge 790. Based on the classification, the sorting module can determine, for example, (i) when to activate one or more sorting mechanisms at the sorting junction of the cartridge 790 to sort one or more cells of interest, (ii) which sub-channel of a plurality of sub channels to direct each single cell for sorting. In some examples, the sorted cells can be collected for further analysis, e.g., downstream molecular assessment and/or profiling, such as genomics, transcriptomics, proteomics, metabolomics, etc. Any of the methods or platforms disclosed herein can be used as a tool that permits a user to train one or more models (e.g., from the modeling

library) for cell clustering and/or cell classification. For example, a user may provide initial image dataset of a sample to the platform, and the platform may process the initial set of image data. Based on the processing, the platform can determine a number of labels and/or an amount of data that the user needs to train the one or more models, based on the initial image dataset of the sample. In some examples, the platform can determine that the initial set of image data can be insufficient to provide an accurate cell classification or cell morphology map. For example, the platform can plot an initial cell morphology map and recommend to the user the number of labels and/or the amount of data needed to for enhanced processing, classification, and/or sorting, based on proximity (or separability), correlation, or commonality of the datapoints in the map (e.g., whether there is no distinguishable clusters within the map, whether the clusters within the map are too close to each other, etc.). In another example, the platform can allow the user to select different model (e.g., clustering model) or classifier, different combinations of models or classifiers, to re-analyze the initial set of image data.

[0223] Any of the methods or platforms disclosed herein can be used to determine quality or state of the image(s) of the cell, that of the cell, or that of a sample comprising the cell. The quality or state of the cell can be determined at a single cell level. In another example, the quality or state of the cell can be determined at an aggregate level (e.g., as a whole sample, or as a portion of the sample). The quality or state can be determined and reported based on, e.g., a number system (e.g., a number scale from about 1 to about 10, a percentage scale from about 1% to about 100%), a symbolic system, or a color system. For example, the quality or state can be indicative of a preparation or priming condition of the sample (e.g., whether the sample has a sufficient number of cells, whether the sample has too much artifacts, debris, etc.) or indicative of a viability of the sample (e.g., whether the sample has an amount of “dead” cells above a predetermined threshold).

[0224] Any of the methods or platforms disclosed herein can be used to sort cells in silico (e.g., prior to actual sorting of the cells using a microfluidic channel). The in silico sorting can be, e.g., to discriminate among and/or between, e.g., multiple different cell types (e.g., different types of cancer cells, different types of immune cells, etc.), cell states, cell qualities. The methods and platforms disclosed herein can utilize pre-determined morphological properties (e.g., provided in the platform) for the discrimination. In another example, newly abstracted morphological properties can be abstracted (e.g., generated) based on the input data for the discrimination. In some examples, new model(s) and/or classifier(s) can be trained or generated to process the image data. In some examples, the newly abstracted morphological properties can be used to discriminate among and/or between, e.g., multiple different cell types, cell states, cell qualities that are known. In another example, the newly abstracted morphological properties can be used to create new class

(or classifications) to sort the cells (e.g., in silico or via the microfluidic system). The newly abstracted morphological properties as disclosed herein may enhance accuracy or sensitivity of cell sorting (e.g., in silico or via the microfluidic system).

[0225] Subsequent to the in silico sorting of the cells, the actual cell sorting of the cells (e.g., via the microfluidic system or cartridge) based on the in silico sorting can be performed within less than about 1 hours, about 50 minutes, about 40 minutes, about 30 minutes, about 25 minutes, about 20 minutes, about 15 minutes, about 10 minutes, about 9 minutes, about 8 minutes, about 7 minutes, about 6 minutes, about 5 minutes, about 4 minutes, about 3 minutes, about 2 minutes, about 1 minute, about 50 seconds, about 40 seconds, about 30 seconds, about 20 seconds, about 10 seconds, about 5 seconds, about 1 second, or less. In some examples, the in silico sorting and the actual sorting can occur in real-time.

[0226] In any of the methods or platforms disclosed herein, the model(s) and/or classifier(s) can be validated (e.g., for the ability to demonstrate accurate cell classification performance). Non-limiting examples of validation metrics that can be utilized can include, but are not limited to, threshold metrics (e.g., accuracy, F-measure, Kappa, Macro-Average Accuracy, Mean-Class-Weighted Accuracy, Optimized Precision, Adjusted Geometric Mean, Balanced Accuracy, etc.), the ranking methods and metrics (e.g., receiver operating characteristics (ROC) analysis or “ROC area under the curve (ROC AUC)”), and the probabilistic metrics (e.g., root-mean-squared error). For example, the model(s) or classifier(s) can be determined to be balanced or accurate when the ROC AUC is greater than 0.5, greater than about 0.55, greater than about 0.6, greater than about 0.65, greater than about 0.7, greater than about 0.75, greater than about 0.8, greater than about 0.85, greater than about 0.9, greater than about 0.91, greater than about 0.92, greater than about 0.93, greater than about 0.94, greater than about 0.95, greater than about 0.96, greater than about 0.97, greater than about 0.98, greater than about 0.99, or more.

[0227] In any of the methods or platforms disclosed herein, the image(s) of the cell(s) can be obtained when the cell(s) are prepared and diluted in a sample (e.g., a buffer sample). The cell(s) can be diluted, e.g., in comparison to real-life concentrations of the cell in the tissue (e.g., solid tissue, blood, serum, spinal fluid, urine, etc.) to a dilution concentration. The methods or platforms disclosed herein can be compatible with a sample (e.g., a biological sample or derivative thereof) that is diluted by a factor of about 500 to about 1,000,000. The methods or platforms disclosed herein can be compatible with a sample that is diluted by a factor of at least about 500. The methods or platforms disclosed herein can be compatible with a sample that is diluted by a factor of at most about 1,000,000. The methods or platforms disclosed herein can be compatible with a sample that is diluted by a factor of about 500 to about 1,000, about 500 to about 2,000, about 500 to about

5,000, about 500 to about 10,000, about 500 to about 20,000, about 500 to about 50,000, about 500 to about 100,000, about 500 to about 200,000, about 500 to about 500,000, about 500 to about 1,000,000, about 1,000 to about 2,000, about 1,000 to about 5,000, about 1,000 to about 10,000, about 1,000 to about 20,000, about 1,000 to about 50,000, about 1,000 to about 100,000, about 1,000 to about 200,000, about 1,000 to about 500,000, about 1,000 to about 1,000,000, about 2,000 to about 5,000, about 2,000 to about 10,000, about 2,000 to about 20,000, about 2,000 to about 50,000, about 2,000 to about 100,000, about 2,000 to about 200,000, about 2,000 to about 500,000, about 2,000 to about 1,000,000, about 5,000 to about 10,000, about 5,000 to about 20,000, about 5,000 to about 50,000, about 5,000 to about 100,000, about 5,000 to about 200,000, about 5,000 to about 500,000, about 5,000 to about 1,000,000, about 10,000 to about 20,000, about 10,000 to about 50,000, about 10,000 to about 100,000, about 10,000 to about 200,000, about 10,000 to about 500,000, about 10,000 to about 1,000,000, about 20,000 to about 50,000, about 20,000 to about 100,000, about 20,000 to about 200,000, about 20,000 to about 500,000, about 20,000 to about 1,000,000, about 50,000 to about 100,000, about 50,000 to about 200,000, about 50,000 to about 500,000, about 50,000 to about 1,000,000, about 100,000 to about 200,000, about 100,000 to about 500,000, about 100,000 to about 1,000,000, about 200,000 to about 500,000, about 200,000 to about 1,000,000, or about 500,000 to about 1,000,000. The methods or platforms disclosed herein can be compatible with a sample that is diluted by a factor of about 500, about 1,000, about 2,000, about 5,000, about 10,000, about 20,000, about 50,000, about 100,000, about 200,000, about 500,000, or about 1,000,000.

[0228] In any of the methods or platforms disclosed herein, the classifier can generate a prediction probability (e.g., based on the morphological clustering and analysis) that an individual cell or a cluster of cells belongs to a cell class (e.g., within a predetermined cell class provided in the CMA as disclosed herein), e.g., via a reporting module. The reporting module can communicate with the user via a GUI as disclosed herein. In another example, the classifier can generate a prediction vector that an individual cell or a cluster of cells belongs to a plurality of cell classes (e.g., a plurality of all of predetermined cell classes from the CMA as disclosed herein). The vector can be 1D (e.g., a single row of different cell classes), 2D (e.g., two dimensions, such as tissue origin vs. cell type), 3D, etc. In some examples, based on processing and analysis of image data obtained from a sample, the classifier can generate a report showing a composition of the sample, e.g., a distribution of one or more cell types, each cell type indicated with a relative proportion within the sample. Each cell of the sample can also be annotated with a most probable cell type and one or more less probable cell types.

[0229] Any one of the methods and platforms disclosed herein can be capable of processing

image data of one or more cells to generate one or more morphometric maps of the one or more cells. Non-limiting examples of morphometric models can be utilized to analyze one or more images of single cells (or cell clusters) can include, e.g., simple morphometries (e.g., based on lengths, widths, masses, angles, ratios, areas, etc.), landmark-based geometric morphometries (e.g., spatial information, intersections, etc. of one or more components of a cell), procrustes-based geometric morphometries (e.g., by removing non-shape information that is altered by translation, scaling, and/or rotation from the image data), Euclidean distance matrix analysis, diffeomorphometry, and outline analysis. The morphometric map(s) can be multi-dimensional (e.g., 2D, 3D, etc.). The morphometric map(s) can be reported to the user via the GUI.

[0230] Any of the methods or platforms disclosed herein (e.g., the analysis module) can be used to process, analyze, classify, and/or compare two or more samples (e.g., at least about 2, about 3, about 4, about 5, about 6, about 7, about 8, about 9, about 10, or more test samples). The two or more samples can each be analyzed to determine a morphological profile (e.g., a cell morphology map) of each sample. For example, the morphological profiles of the two or more samples can be compared for identifying a disease state of a patient's sample in comparison to a health cohort's sample or a sample of image data representative of a disease of interest. In another example, the morphological profiles of the two or more samples can be compared to monitor a progress of a condition of a subject, e.g., comparing first image data of a first set of cells from a subject before a treatment (e.g., a test drug candidate, chemotherapy, surgical resection of solid tumors, etc.) and second image data of a second set of cells from the subject after the treatment. The second set of cells can be obtained from the subject at least about 1 week, at least about 2 weeks, at least about 3 weeks, at least about 4 weeks, at least about 2 months, or at least about 3 months subsequent to obtaining the first set of cells from the subject. In a different example, the morphological profiles of the two or more samples can be compared to monitor effects of two or more different treatment options (e.g., different test drugs) in two or more different cohorts (e.g., human subjects, animal subjects, or cells being tested in vitro/ex vivo). Accordingly, the systems and methods disclosed herein can be utilized (e.g., using sorting or enrichment of a cell type of interest or a cell exhibiting a characteristic of interest) to select a drug and/or a therapy that yields a desired effect (e.g., a therapeutic effect greater than equal to a threshold value).

[0231] Any of the platforms disclosed herein (e.g., cell analysis platform) can provide an inline end-to-end pipeline solution for continuous labeling and/or sorting of multiple different cell types and/or states based at least in part on (e.g., based solely on) morphological analysis of imaging data provided. A modeling library used by the platform can be scalable for large amount of data, extensible (e.g., one or more models or classifiers modified), and/or generalizable (e.g., more

resistant to data perturbations - such as artifacts, debris, random objects in the background, image/video distortions - between samples). Any of the modeling library can be removed or updated with new model automatically by the machine learning algorithms or artificial intelligence, or by the user.

[0232] Any of the methods and platforms disclosed herein can adjust one or more parameters of the microfluidic system as disclosed herein. As cells are flowing through a flow channel, an imaging module (e.g., sensors, cameras) can capture image(s)/video(s) of the cells and generate new image data. The image data can be processed and analyzed (e.g., in real-time) by the methods and platforms of the present disclosure to train a model (e.g., machine learning model) to determine whether or not one or more parameters of the microfluidic system.

[0233] In some examples, the model(s) can determine that the cells are flowing too fast or too slow, and send an instruction to the microfluidic system to adjust (i) the velocity of the cells (e.g., using adjusting velocity of the fluid medium carrying the cells) and/or (ii) image recording rate of a camera that is capturing images/videos of cells flowing through the flow channel.

[0234] In some examples, the model(s) can determine that the cells are in-focus or out-of-focus in the images/videos, and send an instruction to the microfluidic system to (i) adjust a positioning of the cells within the cartridge (e.g., move the cell towards or away from the center of the flow channel via, for example, hydrodynamic focusing and/or inertial focusing) and/or (ii) adjust a focal length/plane of the camera that is capturing images/videos of cells flowing through the flow channel. Adjusting the focal length/plane can be performed for the same cell that has been analyzed (e.g., adjusting focal length/plane of a camera that is downstream) or a subsequent cell. Adjusting the focal length/plane can enhance clarity or reduce blurriness in the images. The focal length/plane can be adjusted based on a classified type or state of the cell. In some examples, adjusting the focal length/plane can allow enhanced focusing/clarity on all parts of the cell. In some examples, adjusting the focal length/plane can allow enhanced focusing/clarity on different portions (but not all parts) of the cell. Without wishing to be bound by any particular theory, out-of-focus images can be usable for any of the methods disclosed herein to extract morphological feature(s) of the cell that otherwise may not be abstracted from in-focus images, or vice versa. Thus, in some examples, instructing the imaging module to capture both in-focus and out-of-focus images of the cells can enhance accuracy of any of the analysis of cells disclosed herein.

[0235] In another example, the model(s) can send an instruction to the microfluidic system to modify the flow and adjust an angle of the cell relative to the camera, to adjust focus on different portions of the cell or a subsequent cell. Different portions as disclosed herein can comprise an upper portion, a mid portion, a lower portion, membrane, nucleus, mitochondria, etc. of the cell.

[0236] In order to image cells at the right focus (with respect to height or z dimension), what is conventionally done is to calculate the “focus measure” of an image using information theoretic methods like Fourier Transform or Laplace transform.

[0237] In some examples, bi-directional out-of-focus(OOF) images cells (e.g., one or more first images that are OOF in a first direction, and one or more second images that are OOF in as second direction that is different — such as opposite — from the first direction). For example, images that are OOF in two opposite directions can be called “bright OOF” image(s) and “dark OOF” image(s), which can be obtained by changing the z-focus bi-directionally. A classifier as disclosed herein can be trained with a image data comprising both bright OOF image(s) and dark OOF image(s). The trained classifiers can be used to run inferences (e.g., in real-time) on new image data of cells to classify each image as bright OOF image, dark OOF image, and optionally image that is not OOF (e.g., not OOF relative to the bright/dark OOF images). The classifier can also measure a percentage of bright OOF image, a percentage of dark OOF image, or a percentage of both bright and dark OOF images within the image data. For example, if any of the percentage of bright OOF image, the percentage of dark OOF image, or the percentage of both bright and dark OOF images is above a threshold value (e.g., a predetermined threshold value), then the classifier can determine that the imaging device (e.g., by the microfluidic system as disclosed herein) may not be imaging cells at the right focal length/plane. The classifier can instruct the user, via GUI of a user device, to adjust the imaging device’s focal length/plane. In some examples, the classifier can determine, based on analysis of the image data comprising OOF images, direction and degree of adjustment of focal length/plane that can be required to adjust the imaging device, to yield a reduced amount of OOF imaging. In some examples, the classifier and the microfluidic device can be operatively coupled to a machine learning/artificial intelligence controller, such that the focal length/plane of the imaging device can be adjusted automatically upon determination of the classifier.

[0238] A threshold (e.g., a predetermined threshold) of a percentage of OOF images (e.g., bright OOF, dark OOF, or both) can be about 0.1 % to about 20 %. A threshold (e.g., a predetermined threshold) of a percentage of OOF images (e.g., bright OOF, dark OOF, or both) can be at least about 0.1 %. A threshold (e.g., a predetermined threshold) of a percentage of OOF images (e.g., bright OOF, dark OOF, or both) can be at most about 20 %. A threshold (e.g., a predetermined threshold) of a percentage of OOF images (e.g., bright OOF, dark OOF, or both) can be about 0.1 % to about 0.5 %, about 0.1 % to about 1 %, about 0.1 % to about 2 %, about 0.1 % to about 4 %, about 0.1 % to about 6 %, about 0.1 % to about 8 %, about 0.1 % to about 10 %, about 0.1 % to about 15 %, about 0.1 % to about 20 %, about 0.5 % to about 1 %, about 0.5 % to about 2 %, about 0.5 % to about 4 %, about 0.5 % to about 6 %, about 0.5 % to about 8 %, about 0.5 % to about 10

%, about 0.5 % to about 15 %, about 0.5 % to about 20 %, about 1 % to about 2 %, about 1 % to about 4 %, about 1 % to about 6 %, about 1 % to about 8 %, about 1 % to about 10 %, about 1 % to about 15 %, about 1 % to about 20 %, about 2 % to about 4 %, about 2 % to about 6 %, about 2 % to about 8 %, about 2 % to about 10 %, about 2 % to about 15 %, about 2 % to about 20 %, about 4 % to about 6 %, about 4 % to about 8 %, about 4 % to about 10 %, about 4 % to about 15 %, about 4 % to about 20 %, about 6 % to about 8 %, about 6 % to about 10 %, about 6 % to about 15 %, about 6 % to about 20 %, about 8 % to about 10 %, about 8 % to about 15 %, about 8 % to about 20 %, about 10 % to about 15 %, about 10 % to about 20 %, or about 15 % to about 20 %. A threshold (e.g., a predetermined threshold) of a percentage of OOF images (e.g., bright OOF, dark OOF, or both) can be at least about 0.1 %, or at least about 0.5 %, or at least about 1 %, or at least about 2 %, or at least about 4 %, or at least about 6 %, or at least about 8 %, or at least about 10 %, or at least about 15 %, or at least or about 20 %.

[0239] In some examples, the model(s) can determine that images of different modalities are needed for any of the analysis disclosed herein. Images of varying modalities can comprise a bright field image, a dark field image, a fluorescent image (e.g., of cells stained with a dye), an in-focus image, an out-of-focus image, a greyscale image, a monochrome image, a multi-chrome image, etc.

[0240] Any of the models or classifiers disclosed herein can be trained on a set of image data that is annotated with one imaging modality. In another example, the models/classifiers can be trained on set of image data that is annotated with a plurality of different imaging modalities (e.g., about 2, about 3, about 4, about 5, or more different imaging modalities). Any of the models/classifiers disclosed herein can be trained on a set of image data that is annotated with a spatial coordinate indicative of a position or location within the flow channel. Any of the models/classifiers disclosed herein can be trained on a set of image data that is annotated with a timestamp, such that a set of images can be processed based on the time they are taken.

[0241] An image of the image data can be processed in various image processing methods, such as horizontal or vertical image flips, orthogonal rotation, gaussian noise, contrast variation, or noise introduction to mimic microscopic particles or pixel-level aberrations. One or more of the processing methods can be used to generate replicas of the image or analyze the image. In some examples, the image can be processed into a lower-resolution image or a lower-dimension image (e.g., by using one or more deconvolution algorithm).

[0242] In any of the methods disclosed herein, processing an image or video from image data can comprise identifying, accounting for, and/or excluding one or more artifacts from the image/video, either automatically or manually by a user. Upon identification, the artifact(s) can be

fed into any of the models or classifiers, to train image processing or image analysis. The artifact(s) can be accounted for when classifying the type or state of one or more cells in the image/video. The artifact(s) can be excluded from any determination of the type or state of the cell(s) in the image/video. The artifact(s) can be removed in silico by any of the models/classifiers disclosed herein, and any new replica or modified variant of the image/video excluding the artifact(s) can be stored in a database as disclosed herein. The artifact(s) can be, for example, from debris (e.g., dead cells, dust, etc.), optical conditions during capturing the image/video of the cells (e.g., lighting variability, over-saturation, under-exposure, degradation of the light source, etc.), external factors (e.g., vibrations, misalignment of the microfluidic chip relative to the lighting or optical sensor/camera, power surges/fluctuations, etc.), and changes to the microfluidic system (e.g., deformation/shrinkage/expansion of the microfluidic channel or the microfluidic chip as a whole). The artifacts can be known. The artifacts can be unknown, and the models or classifiers disclosed herein can be configured to define one or more parameters of a new artifact, such that the new artifact can be identified, accounted for, and/or excluded in image processing and analysis.

[0243] In some examples, a plurality of artifacts disclosed herein can be identified, accounted for, and/or excluded during image/video processing or analysis. The plurality of artifacts can be weighted the same (e.g., determined to have the same degree of influence on the image/video processing or analysis) or can have different weights (e.g., determined to have different degrees of influence on the image/video processing or analysis). Weight assignments to the plurality of artifacts can be instructed manually by the user or determined automatically by the models/classifiers disclosed herein.

[0244] In some examples, one or more reference images or videos of the flow channel (e.g., with or without any cell) can be stored in a database and used as a frame of reference to help identify, account for, and/or exclude any artifact. The reference image(s)/video(s) can be obtained before use of the microfluidics system. The reference image(s)/video(s) can be obtained during the use of the microfluidics system. The reference image(s)/video(s) can be obtained periodically during the use of the microfluidics system, such as, each time the optical sensor/camera captures at least about 5, at least about 10, at least about 20, at least about 50, at least about 100, at least about 200, at least about 500, at least about 1,000, at least about 2,000, at least about 5,000, at least about 10,000, at least about 20,000, at least about 50,000, or at least about 100,000 images. The reference image(s)/video(s) can be obtained periodically during the use of the microfluidics system, such as, each time the microfluidics system passes at least about 5, at least about 10, at least about 20, at least about 50, at least about 100, at least about 200, at least about 500, at least about 1,000, at least about 2,000, at least about 5,000, at least about 10,000, at least about 20,000, at least about 50,000,

at least about 100,000 cells. The reference image(s)/video(s) can be obtained at landmark periods during the use of the microfluidics system, such as, when the optical sensor/camera captures at least about 5, at least about 10, at least about 20, at least about 50, at least about 100, at least about 200, at least about 500, at least about 1,000, at least 2,000, at least about 5,000, at least about 10,000, at least about 20,000, at least about 50,000, at least about 100,000 images. The reference image(s)/video(s) can be obtained at landmark periods during the use of the microfluidics system, such as, when the microfluidics system passes at least about 5, at least about 10, at least about 20, at least about 50, at least about 100, at least about 200, at least about 500, at least about 1,000, at least about 2,000, at least about 5,000, at least about 10,000, at least about 20,000, at least about 50,000, at least about 100,000 images.

[0245] The method and the platform as disclosed herein can be utilized to process (e.g., modify, analyze, classify) the image data at a rate of about 1,000 images/second to about 100,000,000 images/second. The rate of image data processing can be at least about 1,000 images/second. The rate of image data processing can be at most about 100,000,000 images/second. The rate of image data processing can be about 1,000 images/second to about 5,000 images/second, about 1,000 images/second to about 10,000 images/second, about 1,000 images/second to about 50,000 images/second, about 1,000 images/second to about 100,000 images/second, about 1,000 images/second to about 500,000 images/second, about 1,000 images/second to about 1,000,000 images/second, about 1,000 images/second to about 5,000,000 images/second, about 1,000 images/second to about 10,000,000 images/second, about 1,000 images/second to about 50,000,000 images/second, about 1,000 images/second to about 100,000,000 images/second, about 5,000 images/second to about 10,000 images/second, about 5,000 images/second to about 50,000 images/second, about 5,000 images/second to about 100,000 images/second, about 5,000 images/second to about 500,000 images/second, about 5,000 images/second to about 1,000,000 images/second, about 5,000 images/second to about 5,000,000 images/second, about 5,000 images/second to about 10,000,000 images/second, about 5,000 images/second to about 50,000,000 images/second, about 5,000 images/second to about 100,000,000 images/second, about 10,000 images/second to about 50,000 images/second, about 10,000 images/second to about 100,000 images/second, about 10,000 images/second to about 500,000 images/second, about 10,000 images/second to about 1,000,000 images/second, about 10,000 images/second to about 5,000,000 images/second, about 10,000 images/second to about 10,000,000 images/second, about 10,000 images/second to about 50,000,000 images/second, about 10,000 images/second to about 100,000,000 images/second, about 50,000 images/second to about 100,000 images/second, about 50,000 images/second to about 500,000 images/second, about 50,000 images/second to about

1,000,000 images/second, about 50,000 images/second to about 5,000,000 images/second, about 50,000 images/second to about 10,000,000 images/second, about 50,000 images/second to about 50,000,000 images/second, about 50,000 images/second to about 100,000,000 images/second, about 100,000 images/second to about 500,000 images/second, about 100,000 images/second to about 1,000,000 images/second, about 100,000 images/second to about 5,000,000 images/second, about 100,000 images/second to about 10,000,000 images/second, about 100,000 images/second to about 50,000,000 images/second, about 100,000 images/second to about 100,000,000 images/second, about 500,000 images/second to about 1,000,000 images/second, about 500,000 images/second to about 5,000,000 images/second, about 500,000 images/second to about 10,000,000 images/second, about 500,000 images/second to about 50,000,000 images/second, about 500,000 images/second to about 100,000,000 images/second, about 1,000,000 images/second to about 5,000,000 images/second, about 1,000,000 images/second to about 10,000,000 images/second, about 1,000,000 images/second to about 50,000,000 images/second, about 1,000,000 images/second to about 100,000,000 images/second, about 5,000,000 images/second to about 10,000,000 images/second, about 5,000,000 images/second to about 50,000,000 images/second, about 5,000,000 images/second to about 100,000,000 images/second, or about 50,000,000 images/second to about 100,000,000 images/second. The rate of image data processing can be about 1,000 images/second, about 5,000 images/second, about 10,000 images/second, about 50,000 images/second, about 100,000 images/second, about 500,000 images/second, about 1,000,000 images/second, about 5,000,000 images/second, about 10,000,000 images/second, about 50,000,000 images/second, or about 100,000,000 images/second.

[0246] The method and the platform as disclosed herein can be utilized to process (e.g., modify, analyze, classify) the image data at a rate of about 1,000 cells/second to about 100,000,000 cells/second. The rate of image data processing can be at least about 1,000 cells/second. The rate of image data processing can be at most about 100,000,000 cells/second. The rate of image data processing can be about 1,000 cells/second to about 5,000 cells/second, about 1,000 cells/second to about 10,000 cells/second, about 1,000 cells/second to about 50,000 cells/second, about 1,000 cells/second to about 100,000 cells/second, about 1,000 cells/second to about 500,000 cells/second, about 1,000 cells/second to about 1,000,000 cells/second, about 1,000 cells/second to about 5,000,000 cells/second, about 1,000 cells/second to about 10,000,000 cells/second, about 1,000 cells/second to about 50,000,000 cells/second, about 1,000 cells/second to about 100,000,000 cells/second, about 5,000 cells/second to about 10,000 cells/second, about 5,000 cells/second to

about 50,000 cells/second, about 5,000 cells/second to about 100,000 cells/second, about 5,000 cells/second to about 500,000 cells/second, about 5,000 cells/second to about 1,000,000 cells/second, about 5,000 cells/second to about 5,000,000 cells/second, about 5,000 cells/second to about 10,000,000 cells/second, about 5,000 cells/second to about 50,000,000 cells/second, about 5,000 cells/second to about 100,000,000 cells/second, about 10,000 cells/second to about 50,000 cells/second, about 10,000 cells/second to about 100,000 cells/second, about 10,000 cells/second to about 500,000 cells/second, about 10,000 cells/second to about 1,000,000 cells/second, about 10,000 cells/second to about 5,000,000 cells/second, about 10,000 cells/second to about 10,000,000 cells/second, about 10,000 cells/second to about 50,000,000 cells/second, about 10,000 cells/second to about 100,000,000 cells/second, about 50,000 cells/second to about 100,000 cells/second, about 50,000 cells/second to about 500,000 cells/second, about 50,000 cells/second to about 1,000,000 cells/second, about 50,000 cells/second to about 5,000,000 cells/second, about 50,000 cells/second to about 10,000,000 cells/second, about 50,000 cells/second to about 50,000,000 cells/second, about 50,000 cells/second to about 100,000,000 cells/second, about 100,000 cells/second to about 500,000 cells/second, about 100,000 cells/second to about 1,000,000 cells/second, about 100,000 cells/second to about 5,000,000 cells/second, about 100,000 cells/second to about 10,000,000 cells/second, about 100,000 cells/second to about 50,000,000 cells/second, about 100,000 cells/second to about 100,000,000 cells/second, about 500,000 cells/second to about 1,000,000 cells/second, about 500,000 cells/second to about 5,000,000 cells/second, about 500,000 cells/second to about 10,000,000 cells/second, about 500,000 cells/second to about 50,000,000 cells/second, about 500,000 cells/second to about 100,000,000 cells/second, about 1,000,000 cells/second to about 5,000,000 cells/second, about 1,000,000 cells/second to about 10,000,000 cells/second, about 1,000,000 cells/second to about 50,000,000 cells/second, about 1,000,000 cells/second to about 100,000,000 cells/second, about 5,000,000 cells/second to about 10,000,000 cells/second, about 5,000,000 cells/second to about 50,000,000 cells/second, about 5,000,000 cells/second to about 100,000,000 cells/second, about 10,000,000 cells/second to about 50,000,000 cells/second, about 10,000,000 cells/second to about 100,000,000 cells/second, or about 50,000,000 cells/second to about 100,000,000 cells/second. The rate of image data processing can be about 1,000 cells/second, about 5,000 cells/second, about 10,000 cells/second, about 50,000 cells/second, about 100,000 cells/second, about 500,000 cells/second, about 1,000,000 cells/second, about 5,000,000 cells/second, about 10,000,000 cells/second, about 50,000,000 cells/second, or about 100,000,000 cells/second.

[0247] The method and the platform as disclosed herein can be utilized to process (e.g., modify, analyze, classify) the image data at a rate of about 1,000 datapoints/second to about 100,000,000

datapoints/second. The rate of image data processing can be at least about 1,000 datapoints/second. The rate of image data processing can be at most about 100,000,000 datapoints/second. The rate of image data processing can be about 1,000 datapoints/second to about 5,000 datapoints/second, about 1,000 datapoints/second to about 10,000 datapoints/second, about 1,000 datapoints/second to about 50,000 datapoints/second, about 1,000 datapoints/second to about 100,000 datapoints/second, about 1,000 datapoints/second to about 500,000 datapoints/second, about 1,000 datapoints/second to about 1,000,000 datapoints/second, about 1,000 datapoints/second to about 5,000,000 datapoints/second, about 1,000 datapoints/second to about 10,000,000 datapoints/second, about 1,000 datapoints/second to about 50,000,000 datapoints/second, about 1,000 datapoints/second to about 100,000,000 datapoints/second, about 5,000 datapoints/second to about 10,000 datapoints/second, about 5,000 datapoints/second to about 50,000 datapoints/second, about 5,000 datapoints/second to about 100,000 datapoints/second, about 5,000 datapoints/second to about 500,000 datapoints/second, about 5,000 datapoints/second to about 1,000,000 datapoints/second, about 5,000 datapoints/second to about 5,000,000 datapoints/second, about 5,000 datapoints/second to about 10,000,000 datapoints/second, about 5,000 datapoints/second to about 50,000,000 datapoints/second, about 5,000 datapoints/second to about 100,000,000 datapoints/second, about 10,000 datapoints/second to about 50,000 datapoints/second, about 10,000 datapoints/second to about 100,000 datapoints/second, about 10,000 datapoints/second to about 500,000 datapoints/second, about 10,000 datapoints/second to about 1,000,000 datapoints/second, about 10,000 datapoints/second to about 5,000,000 datapoints/second, about 10,000 datapoints/second to about 10,000,000 datapoints/second, about 10,000 datapoints/second to about 50,000,000 datapoints/second, about 10,000 datapoints/second to about 100,000,000 datapoints/second, about 50,000 datapoints/second to about 100,000 datapoints/second, about 50,000 datapoints/second to about 500,000 datapoints/second, about 50,000 datapoints/second to about 1,000,000 datapoints/second, about 50,000 datapoints/second to about 5,000,000 datapoints/second, about 50,000 datapoints/second to about 10,000,000 datapoints/second, about 50,000 datapoints/second to about 50,000,000 datapoints/second, about 50,000 datapoints/second to about 100,000,000 datapoints/second, about 100,000 datapoints/second to about 500,000 datapoints/second, about 100,000 datapoints/second to about 1,000,000 datapoints/second, about 100,000 datapoints/second to about 5,000,000 datapoints/second, about 100,000 datapoints/second to about 10,000,000 datapoints/second, about 100,000 datapoints/second to about 50,000,000 datapoints/second, about 100,000 datapoints/second to about 100,000,000 datapoints/second, about 500,000 datapoints/second to about 1,000,000 datapoints/second, about 500,000 datapoints/second to about 5,000,000

datapoints/second, about 500,000 datapoints/second to about 10,000,000 datapoints/second, about 500,000 datapoints/second to about 50,000,000 datapoints/second, about 500,000 datapoints/second to about 100,000,000 datapoints/second, about 1,000,000 datapoints/second to about 5,000,000 datapoints/second, about 1,000,000 datapoints/second to about 10,000,000 datapoints/second, about 1,000,000 datapoints/second to about 50,000,000 datapoints/second, about 1,000,000 datapoints/second to about 100,000,000 datapoints/second, about 5,000,000 datapoints/second to about 10,000,000 datapoints/second, about 5,000,000 datapoints/second to about 50,000,000 datapoints/second, about 5,000,000 datapoints/second to about 100,000,000 datapoints/second, about 10,000,000 datapoints/second to about 50,000,000 datapoints/second, about 10,000,000 datapoints/second to about 100,000,000 datapoints/second, or about 50,000,000 datapoints/second to about 100,000,000 datapoints/second. The rate of image data processing can be about 1,000 datapoints/second, about 5,000 datapoints/second, about 10,000 datapoints/second, about 50,000 datapoints/second, about 100,000 datapoints/second, about 500,000 datapoints/second, about 1,000,000 datapoints/second, about 5,000,000 datapoints/second, about 10,000,000 datapoints/second, about 50,000,000 datapoints/second, or about 100,000,000 datapoints/second.

[0248] Any of the methods or platforms disclosed herein can be operatively coupled to an online crowdsourcing platform. The online crowdsourcing platform can comprise any of the database disclosed herein. For example, the database can store a plurality of single cell images that are grouped into morphologically-distinct clusters corresponding to a plurality of cell classes (e.g., predetermined cell types or states). The online crowdsourcing platform can comprise one or more models or classifiers as disclosed herein (e.g., a modeling library comprising one or more machine learning models/classifiers as disclosed herein). The online crowdsourcing platform can comprise a web portal for a community of users to share contents, e.g., (1) upload, download, search, curate, annotate, or edit one or more existing images or new images into the database, (2) train or validate the one or more model(s)/classifier(s) using datasets from the database, and/or (3) upload new models into the modeling library. In some examples, the online crowdsourcing platform can allow users to buy, sell, share, or exchange the model(s)/classifier(s) with one another.

[0249] In some examples, the web portal can be configured to generate incentives for the users to update the database with new annotated cell images, model(s), and/or classifier(s). Incentives can be monetary. Incentives can be additional access to the global CMA, model(s), and/or classifier(s). In some examples, the web portal can be configured to generate incentives for the users to download, use, and review (e.g., rate or leave comments) any of the annotated cell images, model(s), and/or classifier(s) from, e.g., other users.

[0250] In some examples, a global cell morphology atlas (global CMA) can be generated using collecting (i) annotated cell images, (ii) cell morphology maps or ontologies, (iii), and/or (iv) classifiers from the users using the web portal. The global CMA can then be shared with the users via the web portal. All users can have access to the global CMA. In another example, specifically defined users can have access to specifically defined portions of the global CMA. For example, cancer centers can have access to “cancer cells” portion of the global CMA, e.g., using a subscription based service. In a similar fashion, global models or classifiers can be generated based on the annotated cell images, model(s), and/or classifiers that are collected from the users using the web portal.

Microfluidic Systems and Methods Thereof

[0251] FIG. 8A shows a schematic illustration of the cell sorting system, as disclosed herein, with a cartridge design (e.g., a microfluidic design), with further details illustrated in FIG. 8B. The cell sorting system can be operatively coupled to a machine learning or artificial intelligence controller. Such ML/AI controller can be configured to perform any of the methods disclosed herein. Such ML/AI controller can be operatively coupled to any of the platforms disclosed herein. In operation, a sample 802 is prepared and injected by a pump 804 (e.g., a syringe pump) into a cartridge 805, or flow-through device. In some examples, the cartridge 805 is a microfluidic device. Although FIG. 8A illustrates a classification and/or sorting system utilizing a syringe pump, any of a number of perfusion systems can be used such as (but not limited to) gravity feeds, peristalsis, or any of a number of pressure systems. In some examples, the sample is prepared by fixation and staining. In some examples, the sample comprises live cells. As can readily be appreciated, the specific manner in which the sample is prepared is largely dependent upon the requirements of a specific application.

[0252] Examples of the pump or other suitable flow unit may be, but are not limited to, a syringe pump, a vacuum pump, an actuator (e.g., linear, pneumatic, hydraulic, etc.), a compressor, or any other suitable device to exert pressure (positive, negative, alternating thereof, etc.) to a fluid that may or may not comprise one or more particles (e.g., one or more cells to be classified, sorted, and/or analyzed). The pump or other suitable flow unit may be configured to raise, compress, move, and/or transfer fluid into or away from the microfluidic channel. In some examples, the pump or other suitable flow unit may be configured to deliver positive pressure, alternating positive pressure and vacuum pressure, negative pressure, alternating negative pressure and vacuum pressure, and/or only vacuum pressure. The cartridge of the present disclosure may comprise (or otherwise be in operable communication with) at least about 1 - e.g., at least about 2, at least about

3, at least about 4, at least about 5, at least about 6, at least about 7, at least about 8, at least about 9, at least about 10, or more, pumps or other flow units. The flow cell may comprise at most about 10, at most about 9, at most about 8, at most about 7, at most about 6, at most about 5, at most about 4, at most about 3, at most about 2, or at most about 1 pumps or other suitable flow units.

[0253] Each pump or other suitable flow unit can be in fluid communication with at least about 1, at least about 2, at least about 3, at least about 4, at least about 5, at least about 6, at least about 7, at least about 8, at least about 9, at least about 10, or more sources of fluid. Each flow unit may be in fluid communication with at most about 10, at most about 9, at most about 8, at most about 7, at most about 6, at most about 5, at most about 4, at most about 3, at most about 2, or at most about 1 fluid. The fluid may contain the particles (e.g., cells). In another example, the fluid may be particle-free. The pump or other suitable flow unit may be configured to maintain, increase, and/or decrease a flow velocity of the fluid within the microfluidic channel of the flow unit. Thus, the pump or other suitable flow unit may be configured to maintain, increase, and/or decrease a flow velocity (e.g., downstream of the microfluidic channel) of the particles. The pump or other suitable flow unit may be configured to accelerate or decelerate a flow velocity of the fluid within the microfluidic channel of the flow unit, thereby accelerating or decelerating a flow velocity of the particles.

[0254] The fluid can be liquid or gas (e.g., air, argon, nitrogen, etc.). The liquid can be an aqueous solution (e.g., water, buffer, saline, etc.). In another example, the liquid can be oil. In some examples, only one or more aqueous solutions can be directed through the microfluidic channels. In another example, only one or more oils can be directed through the microfluidic channels. In another alternative, both aqueous solution(s) and oil(s) can be directed through the microfluidic channels. In some examples, (i) the aqueous solution may form droplets (e.g., emulsions containing the particles) that are suspended in the oil, or (ii) the oil may form droplets (e.g., emulsions containing the particles) that are suspended in the aqueous solution. As can readily be appreciated, any perfusion system, including but not limited to peristalsis systems and gravity feeds, appropriate to a given classification and/or sorting system can be utilized.

[0255] As noted above, the cartridge 805 can be implemented as a fluidic device that focuses cells from the sample into a single streamline that is imaged continuously. In the illustrated example, the cell line is illuminated by a light source 806 (e.g., a lamp, such as an arc lamp) and an optical system 810 that directs light onto an imaging region 838 of the cartridge 805. An objective lens system 812 magnifies the cells by directing light toward the sensor of a high-speed camera system 814.

[0256] In some examples, a 10x, 20x, 40x, 60x, 80x, 100x, or 200x objective is used to magnify

the cells. In some examples, a 10x objective is used to magnify the cells. In some examples, a 20x objective is used to magnify the cells. In some examples, a 40x objective is used to magnify the cells. In some examples, a 60x objective is used to magnify the cells. In some examples, a 80x objective is used to magnify the cells. In some examples, a 100x objective is used to magnify the cells. In some examples, a 200x objective is used to magnify the cells. In some examples, a 10x to a 200x objective is used to magnify the cells, for example a 10x-20x, a 10x-40x, a 10x-60x, a 10x-80x, or 10x-100x objective is used to magnify the cells. As can readily be appreciated by a person having ordinary skill in the art, the specific magnification utilized can vary greatly and is largely dependent upon the requirements of a given imaging system and cell types of interest.

[0257] In some examples, one or more imaging devices can be used to capture images of the cell. In some examples, the imaging device is a high-speed camera. In some examples, the imaging device is a high-speed camera with a micro-second exposure time. In some instances, the exposure time is about 1 millisecond. In some instances, the exposure time is between about 1 millisecond (ms) and about 0.75 millisecond. In some instances, the exposure time is between about 1 ms and about 0.50 ms. In some instances, the exposure time is between about 1 ms and about 0.25 ms. In some instances, the exposure time is between about 0.75 ms and about 0.50 ms. In some instances, the exposure time is between about 0.75 ms and about 0.25 ms. In some instances, the exposure time is between about 0.50 ms and about 0.25 ms. In some instances, the exposure time is between about 0.25 ms and about 0.1 ms. In some instances, the exposure time is between about 0.1 ms and about 0.01 ms. In some instances, the exposure time is between about 0.1 ms and about 0.001 ms. In some instances, the exposure time is between about 0.1 ms and about 1 microsecond (ps). In some examples, the exposure time is between about 1 ps and about 0.1 ps. In some examples, the exposure time is between about 1 ps and about 0.01 ps. In some examples, the exposure time is between about 0.1 ps and about 0.01 ps. In some examples, the exposure time is between about 1 ps and about 0.001 ps. In some examples, the exposure time is between about 0.1 ps and about 0.001 ps. In some examples, the exposure time is between about 0.01 ps and about 0.001 ps.

[0258] In some examples, the cartridge 805 may comprise at least about 1 – e.g., at least about 2, at least about 3, at least about 4, at least about 5, at least about 6, at least about 7, at least about 8, at least about 9, at least about 10, or more, imaging devices (e.g., the high-speed camera system 814) on or adjacent to the imaging region 838. In some examples, the cartridge may at most about 10 – e.g., at most about 9, at most about 8, at most about 7, at most about 6, at most about 5, at most about 4, at most about 3, at most about 2, or at most about 1 imaging device on or adjacent to the imaging region 838. In some examples, the cartridge 805 may comprise a plurality of imaging devices. Each of the plurality of imaging devices may use light from a same light source.

In another example, each of the plurality of imaging devices may use light from different light sources. The plurality of imaging devices can be configured in parallel and/or in series with respect to one another. The plurality of imaging devices can be configured on one or more sides (e.g., two adjacent sides or two opposite sides) of the cartridge 805. The plurality of imaging devices can be configured to view the imaging region 838 along a same axis or different axes with respect to (i) a length of the cartridge 805 (e.g., a length of a straight channel of the cartridge 805) or (ii) a direction of migration of one or more particles (e.g., one or more cells) in the cartridge 805.

[0259] One or more imaging devices of the present disclosure can be stationary while imaging one or more cells, e.g., at the imaging region 838. In another example, one or more imaging devices may move with respect to the flow channel (e.g., along the length of the flow channel, towards and/or away from the flow channel, tangentially about the circumference of the flow channel, etc.) while imaging the one or more cells. In some examples, the one or more imaging devices can be operatively coupled to one or more actuators, such as, for example, a stepper actuator, linear actuator, hydraulic actuator, pneumatic actuator, electric actuator, magnetic actuator, and mechanical actuator (e.g., rack and pinion, chains, etc.).

[0260] In some examples, the cartridge 805 may comprise at least about 1 – e.g., at least about 2, about 3, about 4, about 5, about 6, about 7, about 8, about 9, about 10, or more, imaging regions (e.g., the imaging region 838). In some examples, the cartridge 805 may comprise at most about 10 – e.g., at most about 9, about 8, about 7, about 6, about 5, about 4, about 3, about 2, or about 1 imaging region. In some examples, the cartridge 815 may comprise a plurality of imaging regions, and the plurality of imaging regions can be configured in parallel and/or in series with respect to each another. The plurality of imaging regions may or may not be in fluid communication with each other. In an example, a first imaging region and a second imaging region can be configured in parallel, such that a first fluid that passes through the first imaging region does not pass through a second imaging region. In another example, a first imaging region and a second imaging region can be configured in series, such that a first fluid that passes through the first imaging region also passes through the second imaging region.

[0261] The imaging device(s) (e.g., the high-speed camera) of the imaging system can comprise an electromagnetic radiation sensor (e.g., IR sensor, color sensor, etc.) that detects at least a portion of the electromagnetic radiation that is reflected by and/or transmitted from the cartridge or any content (e.g., the cell) in the cartridge. The imaging device can be in operative communication with one or more sources (e.g., at least about 1, about 2, about 3, about 4, about 5, or more) of the electromagnetic radiation. The electromagnetic radiation can comprise one or more wavelengths from the electromagnetic spectrum including, but not limited to x-rays (about 0.1 nanometers (nm)

to about 10.0 nm; or about 10^{18} Hertz (Hz) to about 10^{16} Hz), ultraviolet (UV) rays (about 10.0 nm to about 380 nm; or about 8×10^{16} Hz to about 10^{15} Hz), visible light (about 380 nm to about 750 nm; or about 8×10^{14} Hz to about 4×10^{14} Hz), infrared (IR) light (about 750 nm to about 0.1 centimeters (cm); or about 4×10^{14} Hz to about 5×10^{11} Hz), and microwaves (about 0.1 cm to about 100 cm; or about 10^8 Hz to about 5×10^{11} Hz). In some examples, the source(s) of the electromagnetic radiation can be ambient light, and thus the cell sorting system may not have an additional source of the electromagnetic radiation.

[0262] The imaging device(s) can be configured to take a two-dimensional image (e.g., one or more pixels) of the cell and/or a three-dimensional image (e.g., one or more voxels) of the cell.

[0263] As can readily be appreciated, the exposure times can differ across different systems and can largely be dependent upon the requirements of a given application or the limitations of a given system such as but not limited to flow rates. Images are acquired and can be analyzed using an image analysis algorithm.

[0264] In some examples, the images are acquired and analyzed post-capture. In some examples, the images are acquired and analyzed in real-time continuously. Using object tracking software, single cells can be detected and tracked while in the field of view of the camera.

[0265] Background subtraction can then be performed. In a number of examples, the cartridge 806 causes the cells to rotate as they are imaged, and multiple images of each cell are provided to a computing system 816 for analysis. In some examples, the multiple images comprise images from a plurality of cell angles.

[0266] The flow rate and channel dimensions can be determined to obtain multiple images of the same cell from a plurality of different angles (i.e., a plurality of cell angles). A degree of rotation between an angle to the next angle can be uniform or non-uniform. In some examples, a full 360° view of the cell is captured. In some examples, 4 images are provided in which the cell rotates 90° between successive frames. In some examples, 8 images are provided in which the cell rotates 45° between successive frames. In some examples, 24 images are provided in which the cell rotates 15° between successive frames. In some examples, at least three or more images are provided in which the cell rotates at a first angle between a first frame and a second frame, and the cell rotates at a second angle between the second frame and a third frame, wherein the first and second angles are different. In some examples, less than the full 360° view of the cell can be captured, and a resulting plurality of images of the same cell can be sufficient to classify the cell (e.g., determine a specific type of the cell).

[0267] The cell can have a plurality of sides. The plurality of sides of the cell can be defined with respect to a direction of the transport (flow) of the cell through the channel. In some examples,

the cell can comprise a stop side, a bottom side that is opposite the top side, a front side (e.g., the side towards the direction of the flow of the cell), a rear side opposite the front side, a left side, and/or a right side opposite the left side. In some examples, the image of the cell can comprise a plurality of images captured from the plurality of angles, wherein the plurality of images comprise: (1) an image captured from the top side of the cell, (2) an image captured from the bottom side of the cell, (3) an image captured from the front side of the cell, (4) an image captured from the rear side of the cell, (5) an image captured from the left side of the cell, and/or (6) an image captured from the right side of the cell.

[0268] In some examples, a two-dimensional “hologram” of a cell can be generated using superimposing the multiple images of the individual cell. The “hologram” can be analyzed to automatically classify characteristics of the cell based upon features including but not limited to the morphological features of the cell.

[0269] In some examples, at least about 1, at least about 2, at least about 3, at least about 4, at least about 5, at least about 6, at least about 7, at least about 8, at least about 9, or at least about 10 images are captured for each cell. In some examples, about 5 or more images are captured for each cell. In some examples, from about 5 to about 10 images are captured for each cell. In some examples, 10 or more images are captured for each cell. In some examples, from about 10 to about 20 images are captured for each cell. In some examples, about 20 or more images are captured for each cell. In some examples, from about 20 to about 50 images are captured for each cell. In some examples, about 50 or more images are captured for each cell. In some examples, from about 50 to about 100 images are captured for each cell. In some examples, at least about 1, at least about 2, at least about 3, at least about 4, at least about 5, at least about 6, at least about 7, at least about 8, at least about 9, at least about 10, at least about 15, at least about 20, at least about 30, at least about 40, at least about 50, or more images may be captured for each cell at a plurality of different angles. In some examples, at most 50, at most about 40, at most about 30, at most about 20, at most about 15, at most about 10, at most about 9, at most about 8, at most about 7, at most about 6, at most about 5, at most about 4, at most about 3, or at most about 2 images can be captured for each cell at a plurality of different angles.

[0270] In some examples, the imaging device is moved so as to capture multiple images of the cell from a plurality of angles. In some examples, the images are captured at an angle between 0 and 90 degrees to the horizontal axis. In some examples, the images are captured at an angle between 90 and 180 degrees to the horizontal axis. In some examples, the images are captured at an angle between 180 and 270 degrees to the horizontal axis. In some examples, the images are captured at an angle between 270 and 360 degrees to the horizontal axis. In some examples,

multiple imaging devices (for e.g. multiple cameras) are used wherein each device captures an image of the cell from a specific cell angle. In some examples, at least about 2, at least about 3, at least about 4, at least about 5, at least about 6, at least about 7, at least about 8, at least about 9, or at least about 10 cameras are used. In some examples, more than about 10 cameras are used, wherein each camera images the cell from a specific cell angle.

[0271] As can readily be appreciated, the number of images that are captured is dependent upon the requirements of a given application or the limitations of a given system. In several examples, the cartridge has different regions to focus, order, and/or rotate cells. Although the focusing regions, ordering regions, and cell rotating regions are discussed as affecting the sample in a specific sequence, a person having ordinary skill in the art would appreciate that the various regions can be arranged differently, where the focusing, ordering, and/or rotating of the cells in the sample can be performed in any order. Regions within a microfluidic device implemented in accordance with an example of the disclosure are illustrated in FIG. 8B. Cartridge 805 may include a filtration region 830 to prevent channel clogging by aggregates/debris or dust particles. Cells pass through a focusing region 832 that focuses the cells into a single streamline of cells that are then spaced by an ordering region 834. In some examples, the focusing region utilizes “inertial focusing” to form the single streamline of cells. In some examples, the focusing region utilizes “hydrodynamic focusing” to focus the cells into the single streamline of cells. Optionally, prior to imaging, rotation can be imparted upon the cells by a rotation region 836. The optionally spinning cells can then pass through an imaging region 838 in which the cells are illuminated for imaging prior to exiting the cartridge. These various regions are described and discussed in further detail below. In some examples, the rotation region 836 may precede the imaging region 838. In some examples, the rotation region 836 can be a part (e.g., a beginning portion, a middle portion, and/or an end portion with respect to a migration of a cell within the cartridge) of the imaging region 838. In some examples, the imaging region 838 can be a part of the rotation region 836.

[0272] In some examples, a single cell is imaged in a field of view of the imaging device, e.g. camera. In some examples, multiple cells are imaged in the same field of view of the imaging device. In some examples, at least about 1, at least about 2, at least about 3, at least about 4, at least about 5, at least about 6, at least about 7, at least about 8, at least about 9, or at least about 10 cells are imaged in the same field of view of the imaging device. In some examples, up to about 100 cells are imaged in the same field of view of the imaging device. In some instances, about 10 to about 100 cells are imaged in the field of view, for example, about 10 to 20 cells, about 10 to about 30 cells, about 10 to about 40 cells, about 10 to about 50 cells, about 10 to about 60 cells, about 10 to about 80 cells, about 10 to about 90 cells, about 20 to about 30 cells, about 20 to about 40 cells,

about 20 to about 50 cells, about 20 to about 60 cells, about 20 to about 70 cells, about 20 to about 80 cells, about 20 to about 90 cells, about 30 to about 40 cells, about 40 to about 50 cells, about 40 to about 60 cells, about 40 to about 70 cells, about 40 to about 80 cells, about 40 to about 90 cells, about 50 to about 60 cells, about 50 to about 70 cells, about 50 to about 80 cells, about 50 to about 90 cells, about 60 to about 70 cells, about 60 to about 80 cells, about 60 to about 90 cells, about 70 to about 80 cells, about 70 to about 90 cells, or about 90 to about 100 cells are imaged in the same field of view of the imaging device.

[0273] In some examples, only a single cell can be allowed to be transported across a cross-section of the flow channel perpendicular to the axis of the flow channel. In some examples, a plurality of cells (e.g., at least about 2, about 3, about 4, about 5, or more cells; at most about 5, about 4, about 3, about 2, or about 1 cell) can be allowed to be transported simultaneously across the cross-section of the flow channel perpendicular to the axis of the flow channel. In such a example, the imaging device (or the processor operatively linked to the imaging device) can be configured to track each of the plurality of cells as they are transported along the flow channel.

[0274] The imaging system can include, among other things, a camera, an objective lens system and a light source. In a number of examples, cartridges similar to those described above can be fabricated using standard 2D microfluidic fabrication techniques, requiring minimal fabrication time and cost.

[0275] Although specific classification and/or sorting systems, cartridges, and microfluidic devices are described above with respect to FIGS. 8A-8B, classification and/or sorting systems can be implemented in any of a variety of ways appropriate to the requirements of specific applications in accordance with various examples of the disclosure. Specific elements of microfluidic devices that can be utilized in classification and/or sorting systems in accordance with some examples of the disclosure are discussed further below.

[0276] In some examples, examples, the microfluidic system can comprise a microfluidic chip (e.g., comprising one or more microfluidic channels for flowing cells) operatively coupled to an imaging device (e.g., one or more cameras). A microfluidic device can comprise the imaging device, and the chip can be inserted into the device, to align the imaging device to an imaging region of a channel of the chip. To align the chip to the precise location for the imaging, the chip can comprise one or more positioning identifiers (e.g., pattern(s), such as numbers, letters, symbols, or other drawings) that can be imaged to determine the positioning of the chip (and thus the imaging region of the channel of the chip) relative to the device as a whole or relative to the imaging device. For image-based alignment (e.g., auto-alignment) of the chip within the device, one or more images of the chip can be captured upon its coupling to the device, and the image(s) can be analyzed by any

of the methods disclosed herein (e.g., using any model or classifier disclosed herein) to determine a degree or score of chip alignment. The positioning identifier(s) can be a “guide” to navigate the stage holding the chip within the device to move within the device towards a correct position relative to the imaging unit.

[0277] In some examples, rule-based image processing can be used to navigate the stage to a precise range of location or a precise location relative to the image unit.

[0278] In some examples, machine learning/artificial intelligence methods as disclosed herein can be modified or trained to identify the pattern on the chip and navigate the stage to the precise imaging location for the image unit, to increase resilience.

[0279] In some examples, machine learning/artificial intelligence methods as disclosed herein can be modified or trained to implement reinforcement learning based alignment and focusing. The alignment process for the chip to the instrument or the image unit can involve moving the stage holding the chip in, e.g., either X or Y axis and/or moving the imaging plane on the Z axis. In the training process, (i) the chip can start at a X, Y, and Z position (e.g., randomly selected), (ii) based on one or more image(s) of the chip and/or the stage holding the chip, a model can determine a movement vector for the stage and a movement for the imaging plane, (iii) depending on whether such movement vector may take the chip closer to the optimum X, Y, and Z position relative to the image unit, an error term can be determined as a loss for the model, and (iv) the magnitude of the error can be either constant or be proportional to how far the current X, Y, and Z position is from an optimal X, Y, and Z position (e.g., can be predetermined). Such trained model can be used to determine, for example, the movement vector and/or movement of the movement for the imaging plane, to enhance relative alignment between the chip and the image unit (e.g., one or more sensors). The alignment can occur subsequent to capturing of the image(s). In another example, the alignment can occur real-time while capturing images/videos of the positioning identifier(s) of the chip.

[0280] One or more flow channels of the cartridge of the present disclosure may have various shapes and sizes. For example, referring to FIGS. 8A-8B, at least a portion of the flow channel (e.g., the focusing region 832, the ordering region 834, the rotation region 836, the imaging region 838, connecting region therebetween, etc.) may have a cross-section that is circular, triangular, square, rectangular, pentagonal, hexagonal, or any partial shape or combination of shapes thereof.

[0281] In some examples, the system of the present disclosure comprises straight channels with rectangular or square cross-sections. In some examples, the system of the present disclosure comprises straight channels with round cross-sections. In some examples, the system comprises straight channels with half-ellipsoid cross-sections. In some examples, the system comprises spiral

channels. In some examples, the system comprises round channels with rectangular cross-sections. In some examples, the system comprises round channels with rectangular channels with round cross-sections. In some examples, the system comprises round channels with half-ellipsoid cross-sections. In some examples, the system comprises channels that are expanding and contracting in width with rectangular cross-sections. In some examples, the system comprises channels that are expanding and contracting in width with round cross-sections. In some examples, the system comprises channels that are expanding and contracting in width with half-ellipsoid cross-sections.

[0282] The flow channel can comprise one or more walls that are formed to focus one or more cells into a streamline. The flow channel can comprise a focusing region comprising the wall(s) to focus the cell(s) into the streamline. Focusing regions on a microfluidic device can take a disorderly stream of cells and utilize a variety of forces (for e.g. inertial lift forces (wall effect and shear gradient forces) or hydrodynamic forces) to focus the cells within the flow into a streamline of cells. In some examples, the cells are focused in a single streamline. In some examples, the cells are focused in multiple streamlines, for example at least about 2, at least about 3, at least about 4, at least about 5, at least about 6, at least about 7, at least about 8, at least about 9, or at least about 10 streamlines.

[0283] The focusing region receives a flow of randomly arranged cells using an upstream section. The cells flow into a region of contracted and expanded sections in which the randomly arranged cells are focused into a single streamline of cells. The focusing can be driven by the action of inertial lift forces (wall effect and shear gradient forces) acting on cells.

[0284] In some examples, the focusing region is formed with curvilinear walls that form periodic patterns. In some examples, the patterns form a series of square expansions and contractions. In other examples, the patterns are sinusoidal. In further examples, the sinusoidal patterns are skewed to form an asymmetric pattern. The focusing region can be effective in focusing cells over a wide range of flow rates. In the illustrated example, an asymmetrical sinusoidal-like structure is used as opposed to square expansions and contractions. This helps prevent the formation of secondary vortices and secondary flows behind the particle flow stream. In this way, the illustrated structure allows for faster and more accurate focusing of cells to a single lateral equilibrium position. Spiral and curved channels can also be used in an inertia regime; however, these can complicate the integration with other modules. Finally, straight channels where channel width is greater than channel height can also be used for focusing cells onto single lateral position. However, in this case, since there will be more than one equilibrium position in the z-plane, imaging can become problematic, as the imaging focal plane is preferably fixed. As can readily be appreciated, any of a variety of structures that provide a cross section that expands and

contracts along the length of the microfluidic channel or are capable of focusing the cells can be utilized as appropriate to the requirements of specific applications.

[0285] The cell sorting system can be configured to focus the cell at a width and/or a height within the flow channel along an axis of the flow channel. The cell can be focused to a center or off the center of the cross-section of the flow channel. The cell can be focused to a side (e.g., a wall) of the cross-section of the flow channel. A focused position of the cell within the cross-section of the channel can be uniform or non-uniform as the cell is transported through the channel.

[0286] While specific implementations of focusing regions within microfluidic channels are described above, any of a variety of channel configurations that focus cells into a single streamline can be utilized as appropriate to the requirements of a specific application in accordance with various examples of the disclosure.

[0287] Microfluidic channels can be designed to impose ordering upon a single streamline of cells formed by a focusing region in accordance with several examples of the disclosure. Microfluidic channels in accordance with some examples of the disclosure include an ordering region having pinching regions and curved channels. The ordering region orders the cells and distances single cells from each other to facilitate imaging. In some examples, ordering is achieved by forming the microfluidic channel to apply inertial lift forces and Dean drag forces on the cells.

[0288] Different geometries, orders, and/or combinations can be used. In some examples, pinching regions can be placed downstream from the focusing channels without the use of curved channels. Adding the curved channels can help with more rapid and controlled ordering, as well as increasing the likelihood that particles follow a single lateral position as they migrate downstream. As can readily be appreciated, the specific configuration of an ordering region is largely determined based upon the requirements of a given application.

[0289] Architecture of the microfluidic channels of the cartridge of the present disclosure can be controlled (e.g., modified, optimized, etc.) to modulate cell flow along the microfluidic channels. Examples of the cell flow may include (i) cell focusing (e.g., into a single streamline) and (ii) rotation of the one or more cells as the cell(s) are migrating (e.g., within the single streamline) down the length of the microfluidic channels. In some examples, microfluidic channels can be configured to impart rotation on ordered cells in accordance with a number of examples of the disclosure. One or more cell rotation regions (e.g., the cell rotation region 836) of microfluidic channels in accordance with some examples of the disclosure use co-flow of a particle-free buffer to induce cell rotation by using the co-flow to apply differential velocity gradients across the cells. In some examples, a cell rotation region may introduce co-flow of at least about 1, about 2, about 3, about 4, about 5, or more buffers (e.g., particle-free, or containing one or more particles, such

as polymeric or magnetic particles) to impart rotation on one or more cells within the channel. In some examples, a cell rotation region may introduce co-flow of at most about 5, about 4, about 3, about 2, or about 1 buffer to impart the rotation of one or more cells within the channel. In some examples, the plurality of buffers can be co-flown at a same position along the length of the cell rotation region, or sequentially at different positions along the length of the cell rotation region. In some examples, the plurality of buffers can be the same or different. In several examples, the cell rotation region of the microfluidic channel is fabricated using a two-layer fabrication process so that the axis of rotation is perpendicular to the axis of cell downstream migration and parallel to cell lateral migration.

[0290] Cells can be imaged in at least a portion of the cell rotating region, while the cells are tumbling and/or rotating as they migrate downstream. In another example, the cells can be imaged in an imaging region that is adjacent to or downstream of the cell rotating region. In some examples, the cells can be flowing in a single streamline within a flow channel, and the cells can be imaged as the cells are rotating within the single streamline. A rotational speed of the cells can be constant or varied along the length of the imaging region. This may allow for the imaging of a cell at different angles (e.g., from a plurality of images of the cell taken from a plurality of angles due to rotation of the cell), which may provide more accurate information concerning cellular features than can be captured in a single image or a sequence of images of a cell that is not rotating to any significant extent. This also allow a 3D reconstruction of the cell using available software since the angles of rotation across the images are known. In another example, every single image of the sequence of image many be analyzed individually to analyze (e.g., classify) the cell from each image. In some examples, results of the individual analysis of the sequence of images can be aggregated to determine a final decision (e.g., classification of the cell).

[0291] In some examples, a cell rotation region of a microfluidic channel incorporates an injected co-flow prior to an imaging region in accordance with an example of the disclosure. Co-flow can be introduced in the z plane (perpendicular to the imaging plane) to spin the cells. Since the imaging is done in the x-y plane, rotation of cells around an axis parallel to the y-axis provides additional information by rotating portions of the cell that may have been occluded in previous images into view in each subsequent image. Due to a change in channel dimensions, at point x_0 , a velocity gradient is applied across the cells, which can cause the cells to spin. The angular velocity of the cells depends on channel and cell dimensions and the ratio between Q1 (main channel flow rate) and Q2 (co-flow rate) and can be configured as appropriate to the requirements of a given application. In some examples, a cell rotation region incorporates an increase in one dimension of the microfluidic channel to initiate a change in the velocity gradient across a cell to impart rotation

onto the cell. In some examples, a cell rotation region of a microfluidic channel incorporates an increase in the z-axis dimension of the cross section of the microfluidic channel prior to an imaging region in accordance with an example of the disclosure. The change in channel height can initiate a change in velocity gradient across the cell in the z axis of the microfluidic channel, which can cause the cells to rotate as with using co flow.

[0292] In some examples, the system and methods of the present disclosure focuses the cells in microfluidic channels. The term focusing as used herein broadly means controlling the trajectory of cell/cells movement and comprises controlling the position and/or speed at which the cells travel within the microfluidic channels. In some examples controlling the lateral position and/or the speed at which the particles travel inside the microfluidic channels, allows to accurately predict the time of arrival of the cell at a bifurcation. The cells may then be accurately sorted. The parameters critical to the focusing of cells within the microfluidic channels include, but are not limited to channel geometry, particle size, overall system throughput, sample concentration, imaging throughput, size of field of view, and method of sorting.

[0293] In some examples the focusing is achieved using inertial forces. In some examples, the system and methods of the present disclosure focus cells to a certain height from the bottom of the channel using inertial focusing. In these examples, the distance of the cells from the objective is equal and images of all the cells will be clear. As such, cellular details, such as nuclear shape, structure, and size appear clearly in the outputted images with minimal blur. In some examples, the system disclosed herein has an imaging focusing plane that is adjustable. In some examples, the focusing plane is adjusted by moving the objective or the stage. In some examples, the best focusing plane is found by recording videos at different planes and the plane wherein the imaged cells have the highest Fourier magnitude, thus, the highest level of detail and highest resolution, is the best plane.

[0294] In some examples, the system and methods of the present disclosure utilize a hydrodynamic-based z focusing system to obtain a consistent z height for the cells of interests that are to be imaged. In some examples, the design comprises hydrodynamic focusing using multiple inlets for main flow and side flow. In some examples, the hydrodynamic-based z focusing system is a triple-punch design. In some examples, the design comprises hydrodynamic focusing with three inlets, wherein the two side flows pinch cells at the center. For certain channel designs, dual z focus points can be created, wherein a double-punch design similar to the triple-punch design can be used to send objects to one of the two focus points to get consistent focused images. In some examples, the design comprises hydrodynamic focusing with 2 inlets, wherein only one side flow channel is used and cells are focused near channel wall. In some examples, the hydrodynamic

focusing comprises side flows that do not contain any cells and a middle inlet that contains cells. The ratio of the flow rate on the side channel to the flow rate on the main channel determines the width of cell focusing region. In some examples, the design is a combination of the above. In all examples, the design is integrable with the bifurcation and sorting mechanisms disclosed herein. In some examples, the hydrodynamic-based z focusing system is used in conjunction with inertia-based z focusing.

[0295] In some examples, the cell is a live cell. In some examples, the cell is a fixed cell (e.g., in methanol or paraformaldehyde). In some examples, one or more cells can be coupled (e.g., attached covalently or non-covalently) to a substrate (e.g., a polymeric bead or a magnetic bead) while flowing through the cartridge. In some examples, the cell(s) may not be coupled to any substrate while flowing through the cartridge.

[0296] A variety of techniques can be utilized to classify images of cells captured by classification and/or sorting systems in accordance with various examples of the disclosure. In some examples, the image captures are saved for future analysis/classification either manually or by image analysis software. Any suitable image analysis software can be used for image analysis. In some examples, image analysis is performed using OpenCV. In some examples, analysis and classification is performed in real time.

[0297] In some examples, the system and methods of the present disclosure comprise collecting a plurality of images of objects in the flow. In some examples, the plurality of images comprises at least 20 images of cells. In some examples, the plurality of images comprises at least about 19, about 18, about 17, about 16, about 15, about 14, about 13, about 12, about 11, about 10, about 9, about 8, about 7, about 6, about 5, about 4, about 3, or about 2 images of cells. In some examples, the plurality of images comprises images from multiple cell angles. In some examples, the plurality of images, comprising images from multiple cell angles, help derive extra features from the particle which would be hidden if the particle is imaged from a single point-of-view. In some examples, without wishing to be bound by any particular theory, the plurality of images, comprising images from multiple cell angles, help derive extra features from the particle which would be hidden if a plurality of images are combined into a multi-dimensional reconstruction (e.g., a two-dimensional hologram or a three-dimensional reconstruction).

[0298] In some examples, the systems and methods of present disclosure allow for a tracking ability, wherein the system and methods track a particle (e.g., cell) under the camera and maintain the knowledge of which frames belong to the same particle. In some examples, the particle is tracked until it has been classified and/or sorted. In some examples, the particle can be tracked by one or more morphological (e.g., shape, size, area, volume, texture, thickness, roundness, etc.)

and/or optical (e.g., light emission, transmission, reflectance, absorbance, fluorescence, luminescence, etc.) characteristics of the particle. In some examples, each particle can be assigned a score (e.g., a characteristic score) based on the one or more morphological and/or optical characteristics, thereby to track and confirm the particle as the particle travels through the microfluidic channel.

[0299] In some examples, the systems and methods of the disclosure comprise imaging a single particle in a particular field of view of the camera. In some examples, the same instrument that performs imaging operations can also perform sorting operations. In some examples, the system and methods of the present disclosure image multiple particles in the same field of view of camera. Imaging multiple particles in the same field of view of the camera can provide additional advantages, for example it will increase the throughput of the system by batching the data collection and transmission of multiple particles. In some instances, at least about 2, at least about 3, at least about 4, at least about 5, at least about 6, at least about 7, at least about 8, at least about 9, at least about 10, at least about 20, at least about 30, at least about 40, at least about 50, at least about 60, at least about 70, at least about 80, at least about 90, at least about 100, or more particles are imaged in the same field of view of the camera. In some instances, about 100 to about 200 particles are imaged in the same field of view of the camera. In some instances, at most about 100, at most about 90, at most about 80, at most about 70, at most about 60, at most about 50, at most about 40, at most about 30, at most about 20, at most about 10, at most about 9, at most about 8, at most about 7, at most about 6, at most about 5, at most about 4, at most about 3, or at most about 2 particles are imaged in the same field of view of the camera. In some examples, the number of the particles (e.g., cells) that are imaged in the same field of view may not be changed throughout the operation of the cartridge. In another example, the number of the particles (e.g., cells) that are imaged in the same field of view can be changed in real-time throughout the operation of the cartridge, e.g., to increase speed of the classification and/or sorting process without negatively affecting quality or accuracy of the classification and/or sorting process.

[0300] The imaging region maybe downstream of the focusing region and the ordering region. Thus, the imaging region may not be part of the focusing region and the ordering region. In an example, the focusing region may not comprise or be operatively coupled to any imaging device that is configured to capture one or more images to be used for particle analysis (e.g., cell classification).

[0301] In some examples, the systems and the methods of the present disclosure actively sorts a stream of particles. The term sort or sorting as used herein refers to physically separating particles, for e.g. cells, with one or more desired characteristics. The desired characteristic(s) can

comprise a feature of the cell(s) analyzed and/or obtained from the image(s) of the cell.

[0302] Examples of the morphometric feature of the cell(s) can comprise a size, shape, volume, electromagnetic radiation absorbance and/or transmittance (e.g., fluorescence intensity, luminescence intensity, etc.), or viability (e.g., when live cells are used).

[0303] The flow channel can branch into a plurality of channels, and the cell sorting system can be configured to sort the cell by directing the cell to a selected channel of the plurality of channels based on the analyzed image of the cell. The analyzed image can be indicative of one or more features of the cell, wherein the feature(s) are used as parameters of cell sorting. In some examples, one or more channels of the plurality of channels can have a plurality of sub channels, and the plurality of sub-channels can be used to further sort the cells that have been sorted once.

[0304] Cell sorting may comprise isolating one or more target cells from a population of cells. The target cell(s) can be isolated into a separate reservoir that keeps the target cell(s) separate from the other cells of the population. Cell sorting accuracy can be defined as a proportion (e.g., a percentage) of the target cells in the population of cells that have been identified and sorted into the separate reservoir. In some examples, the cell sorting accuracy of the cartridge provided herein can be at least about 80 %, at least about 81 %, at least about 82 %, at least about 83 %, at least about 84 %, at least about 85 %, at least about 86 %, at least about 87 %, at least about 88 %, at least about 89 %, at least about 90 %, at least about 91 %, at least about 92 %, at least about 93 %, at least about 94 %, at least about 95 %, at least about 96 %, at least about 97 %, at least about 98 %, at least about 99 %, or more (e.g., about 99.9% or about 100%). In some examples, the cell sorting accuracy of the cartridge provided herein may be at most about 100 %, at most about 99 %, at most about 98 %, at most about 97 %, at most about 96 %, at most about 95 %, at most about 94 %, at most about 93 %, at most about 92 %, at most about 91 %, at most about 90 %, at most about 89 %, at most about 88 %, at most about 87 %, at most about 86 %, at most about 85 %, at most about 84 %, at most about 83 %, at most about 82 %, at most about 81 %, or at most about 80 %, or less.

[0305] In some examples, cell sorting may be performed at a rate of at least about 1 cell/second, at least about 5 cells/second, at least about 10 cells/second, at least about 50 cells/second, at least about 100 cells/second, at least about 500 cells/second, at least about 1,000 cells/second, at least about 5,000 cells/second, at least about 10,000 cells/second, at least about 50,000 cells/second, or more. In some examples, cell sorting may be performed at a rate of at most about 50,000 cells/second, at most about 10,000 cells/second, at most about 5,000 cells/second, at most about 1,000 cells/second, at most about 500 cells/second, at most about 100 cells/second, at most about 50 cells/second, at most about 10 cells/second, at most about 5 cells/second, or at most about 1

cell/second, or less.

[0306] In some examples, the systems and methods disclosed herein use an active sorting mechanism. In various examples, the active sorting is independent from analysis and decision making platforms and methods. In various examples the sorting is performed by a sorter, which receives a signal from the decision making unit (e.g. a classifier), or any other external unit, and then sorts cells as they arrive at the bifurcation. The term bifurcation as used herein refers to the termination of the flow channel into two or more channels, such that cells with the one or more desired characteristics are sorted or directed towards one of the two or more channels and cell without the one or more desired characteristics are directed towards the remaining channels. In some examples, the flow channel terminates into at least about 2, at least about 3, at least about 4, at least about 5, at least about 6, at least about 7, at least about 8, at least about 9, at least about 10, or more channels. In some examples, the flow channel terminates into at most about 10, at most about 9, at most about 8, at most about 7, at most about 6, at most about 5, at most about 4, at most about 3, or at most about 2 channels. In some examples, the flow channel terminates in two channels and cells with one or more desired characteristics are directed towards one of the two channels (the positive channel), while cells without the one or more desired characteristics are directed towards the other channel (the negative channel). In some examples, the flow channel terminates in three channels and cells with a first desired characteristic are directed to one of the three channels, cells with a second desired characteristic are directed to another of the three channels, and cells without the first desired characteristic and the second desired characteristic are directed to the remaining of the three channels.

[0307] In some examples, the sorting is performed by a sorter. The sorter may function by predicting the exact time at which the particle will arrive at the bifurcation. To predict the time of particle arrival, the sorter can use any applicable method. In some examples, the sorter predicts the time of arrival of the particle by using (i) velocity of particles (e.g., downstream velocity of a particle along the length of the microfluidic channel) that are upstream of the bifurcation and (ii) the distance between velocity measurement/calculation location and the bifurcation. In some examples, the sorter predicts the time of arrival of the particles by using a constant delay time as an input.

[0308] In some examples, prior to the cell's arrival at the bifurcation, the sorter may measure the velocity of a particle (e.g., a cell) at least about 1, at least about 2, at least about 3, at least about 4, or at least about 5, or more times. In some examples, prior to the cell's arrival at the bifurcation, the sorter may measure the velocity of the particle at most about 5, at most about 4, at most about 3, at most about 2, or at most about 1 time. In some examples, the sorter may use at least about 1,

at least about 2, at least about 3, at least about 4, or at least about 5, or more sensors. In some examples, the sorter may use at most about 5, at most about 4, at most about 3, at most about 2, or at most about 1 sensor. Example of the sensor(s) can be an imaging device (e.g., a camera such as a high-speed camera), one- or multi-point light (e.g., laser) detector, etc. Referring to FIGS. 8A-8B the sorter may use any one of the imaging devices (e.g., the high-speed camera system 814) disposed at or adjacent to the imaging region 838. In some examples, the same imaging device(s) can be used to capture one or more images of a cell as the cell is rotating and migrating within the channel, and the one or more images can be analyzed to (i) classify the cell and (ii) measure a rotational and/or lateral velocity of the cell within the channel and predict the cell's arrival time at the bifurcation. In some examples, the sorter may use one or more sensors that are different than the imaging devices of the imaging region 838. The sorter may measure the velocity of the particle (i) upstream of the imaging region 838, (ii) at the imaging region 838, and/or (iii) downstream of the imaging region 838.

[0309] The sorter may comprise or be operatively coupled to a processor, such as a computer processor. Such processor can be the processor 816 that is operatively coupled to the imaging device 814 or a different processor. The processor can be configured to calculate the velocity of a particle (rotational and/or downstream velocity of the particle) and predict the time of arrival of the particle at the bifurcation. The processor can be operatively coupled to one or more valves of the bifurcation. The processor can be configured to direct the valve(s) to open and close any channel in fluid communication with the bifurcation. The processor can be configured to predict and measure when operation of the valve(s) (e.g., opening or closing) is completed.

[0310] In some examples, the sorter may comprise a self-included unit (e.g., comprising the sensors, such as the imaging device(s)) which is capable of (i) predicting the time of arrival of the articles and/or (ii) detecting the particle as it arrives at the bifurcation. In order to sort the particles, the order at which the particles arrive at the bifurcation, as detected by the self-included unit, can be matched to the order of the received signal from the decision making unit (e.g. a classifier). In some examples, controlled particles are used to align and update the order as necessary. In some examples, the decision making unit may classify a first cell, a second cell, and a third cell, respectively, and the sorter may confirm that the first cell, the second cell, and the third cell are sorted, respectively in the same order. If the order is confirmed, the classification and sorting mechanisms (or deep learning algorithms) may remain the same. If the order is different between the classifying and the sorting, then the classification and/or sorting mechanisms (or deep learning algorithms) can be updated or optimized, either manually or automatically. In some examples, the controlled particles can be cells (e.g., live or dead cells).

[0311] In some examples, the controlled particles can be special calibration beads (e.g., plastic beads, metallic beads, magnetic beads, etc.). In some examples the calibration beads used are polystyrene beads with size ranging between about 1 mM to about 50 mM. In some examples the calibration beads used are polystyrene beads with size of least about 1 pM. In some examples the calibration beads used are polystyrene beads with size of at most about 50 pM. In some examples the calibration beads used are polystyrene beads with size ranging between about 1 pM to about 3 pM, about 1 pM to about 5 pM, about 1 pM to about 6 pM, about 1 pM to about 10 pM, about 1 pM to about 15 pM, about 1 pM to about 20 pM, about 1 pM to about 25 pM, about 1 pM to about 30 pM, about 1 pM to about 35 pM, about 1 pM to about 40 pM, about 1 pM to about 50 pM, about 3 pM to about 5 pM, about 3 pM to about 6 pM, about 3 pM to about 10 pM, about 3 pM to about 15 pM, about 3 pM to about 20 pM, about 3 pM to about 25 pM, about 3 pM to about 30 pM, about 3 pM to about 35 pM, about 3 pM to about 40 pM, about 3 pM to about 50 pM, about 5 pM to about 6 pM, about 5 pM to about 10 pM, about 5 pM to about 15 pM, about 5 pM to about 20 pM, about 5 pM to about 25 pM, about 5 pM to about 30 pM, about 5 pM to about 35 pM, about 5 pM to about 40 mM, about 5 mM to about 50 mM, about 6 mM to about 10 mM, about 6 mM to about 15 mM, about 6 mM to about 20 mM, about 6 mM to about 25 mM, about 6 mM to about 30 mM, about 6 mM to about 35 mM, about 6 mM to about 40 mM, about 6 mM to about 50 mM, about 10 mM to about 15 mM, about 10 mM to about 20 mM, about 10 mM to about 25 mM, about 10 mM to about 30 mM, about 10 mM to about 35 mM, about 10 mM to about 40 mM, about 10 mM to about 50 mM, about 15 mM to about 20 mM, about 15 mM to about 25 mM, about 15 mM to about 30 mM, about 15 mM to about 35 mM, about 15 mM to about 40 mM, about 15 mM to about 50 mM, about 20 mM to about 25 mM, about 20 mM to about 30 mM, about 20 mM to about 35 mM, about 20 mM to about 40 mM, about 20 mM to about 50 mM, about 25 mM to about 30 mM, about 25 mM to about 35 mM, about 25 mM to about 40 mM, about 25 mM to about 50 mM, about 30 mM to about 35 mM, about 30 mM to about 40 mM, about 30 mM to about 50 mM, about 35 mM to about 40 mM, about 35 mM to about 50 mM, or about 40 mM to about 50 mM. In some examples the calibration beads used are polystyrene beads with size of about 1 mM, about 3 mM, about 5 mM, about 6 mM, about 10 mM, about 15 mM, about 20 mM, about 25 mM, about 30 mM, about 35 mM, about 40 mM, or about 50 mM.

[0312] In some examples, the sorter (or an additional sensor disposed at or adjacent to the bifurcation) can be configured to validate arrival of the particles (e.g., the cells) at the bifurcation. In some examples, the sorter can be configured to measure an actual arrival time of the particles (e.g., the cells) at the bifurcation. The sorter may analyze (e.g., compare) the predicted arrival time, the actual arrival time, the velocity of the particles downstream of the channel prior to any

adjustment of the velocity, and/or a velocity of the particles downstream of the channel subsequent to such adjustment of the velocity. Based on the analyzing, the sorter may modify any operation (e.g., cell focusing, cell rotation, controlling cell velocity, cell classification algorithms, valve actuation processes, etc.) of the cartridge. The validation by the sorter can be used for closed-loop and real-time update of any operation of the cartridge.

[0313] In some examples, to predict the time of arrival of one or more cells for sorting, the systems, methods, and platforms disclosed herein can dynamically adjust a delay time (e.g., a constant delay time) based on imaging of the cell(s) or based on tracking of the cell(s) with light (e.g., laser). By detecting changes (e.g., flow rates, velocity of aggregate of multiple cells, the lateral location of cells in the channel, etc.) the delay time (e.g., time at which the cells arrive at the bifurcation) can be predicted and adjusted in real-time (e.g., every few milliseconds). A feedback loop can be designed that can constantly read such changes and adjust the delay time accordingly. In another example, the delay time can be adjusted for each cell/particle. The delay time can be calculated separately for each individual cell, based on, e.g., its velocity, lateral position in the channel, and/or time of arrival at specific locations along the channel (e.g., using tracking based on lasers or other methods). The calculated delay time can then be applied to the individual cell/particle (e.g., if the cell is a positive cell or a target cell, the sorting can be performed according to its specific delay time or a predetermined delay time). In some examples, the sorters used in the systems and methods disclosed herein are self-learning cell sorting systems or intelligent cell sorting systems, as disclosed herein.

[0314] These sorting systems can continuously learn based on the outcome of sorting. For example, a sample of cells is sorted, the sorted cells are analyzed, and the results of this analysis are fed back to the classifier. In some examples, the cells that are sorted as “positive” (i.e., target cells or cells of interest) can be analyzed and validated. In some examples, the cells that are sorted as “negative” (i.e., non-target cells or cells not of interest) can be analyzed and validated. In some examples, both positive and negative cells can be validated. Such validation of sorted cells (e.g., based on secondary imaging and classification) can be used for closed-loop and real-time update of the primary cell classification algorithms.

[0315] In some examples, a flush mechanism can be used during sorting. The flush mechanism can ensure that the cell which has been determined to be sorted to a specific bucket or well will end up there (e.g., not be stuck in various parts of the channel or outlet). The flush mechanism can ensure that the channel and outlets stay clean and debris-free for maximum durability. The flush mechanism can inject additional solutions/reagents (e.g., cell lysis buffers, barcoded reagents, etc.) to the well or droplet that the cell is being sorted into. The flush mechanism can be supplied by a

separate set of channels and/or valves which are responsible to flow a fluid at a predefined cadence in the direction of sorting.

[0316] In some examples, the methods and systems disclosed herein can use any sorting technique to sort particles. At least a portion of the collection reservoir may or may not be pre filled with a fluid, e.g., a buffer. In some examples, the sorting technique comprises closing a channel on one side of the bifurcation to collect the desired cell on the other side. In some examples, the closing of the channels can be carried out by employing any known technique. In some examples, the closing is carried out by application of a pressure. In some instances, the pressure is pneumatic actuation. In some examples, the pressure can be positive pressure or negative pressure. In some examples, positive pressure is used. In some examples, one side of the bifurcation is closed by applying pressure and deflecting the soft membrane between top and bottom layers. Other examples of systems and methods of particle (e.g., cell) imaging, analysis, and sorting are further described in International Application No. PCT/US2017/033676 and International Application No. PCT/US2019/046557, each of which is incorporated herein by reference in its entirety.

[0317] In various examples, the systems and methods of the present disclosure comprise one or more reservoirs designed to collect the particles after the particles have been sorted. In some examples, the number of cells to be sorted is about 1 cell to about 1,000,000 cells. In some examples, the number of cells to be sorted is at least about 1 cell. In some examples, the number of cells to be sorted is at most about 1,000,000 cells. In some examples, the number of cells to be sorted is about 1 cell to about 100 cells, about 1 cell to about 500 cells, about 1 cell to about 1,000 cells, about 1 cell to about 5,000 cells, about 1 cell to about 10,000 cells, about 1 cell to about 50,000 cells, about 1 cell to about 100,000 cells, about 1 cell to about 500,000 cells, about 1 cell to about 1,000,000 cells, about 100 cells to about 500 cells, about 100 cells to about 1,000 cells, about 100 cells to about 5,000 cells, about 100 cells to about 10,000 cells, about 100 cells to about 50,000 cells, about 100 cells to about 100,000 cells, about 100 cells to about 500,000 cells, about 100 cells to about 1,000,000 cells, about 500 cells to about 1,000 cells, about 500 cells to about 5,000 cells, about 500 cells to about 10,000 cells, about 500 cells to about 50,000 cells, about 500 cells to about 100,000 cells, about 500 cells to about 500,000 cells, about 500 cells to about 1,000,000 cells, about 1,000 cells to about 5,000 cells, about 1,000 cells to about 10,000 cells, about 1,000 cells to about 50,000 cells, about 1,000 cells to about 100,000 cells, about 1,000 cells to about 500,000 cells, about 1,000 cells to about 1,000,000 cells, about 5,000 cells to about 10,000 cells, about 5,000 cells to about 50,000 cells, about 5,000 cells to about 100,000 cells, about 5,000 cells to about 500,000 cells, about 5,000 cells to about 1,000,000 cells, about 10,000 cells to about 50,000 cells, about 10,000 cells to about 100,000 cells, about 10,000 cells to about 500,000 cells,

about 10,000 cells to about 1,000,000 cells, about 50,000 cells to about 100,000 cells, about 50,000 cells to about 500,000 cells, about 50,000 cells to about 1,000,000 cells, about 100,000 cells to about 500,000 cells, about 100,000 cells to about 1,000,000 cells, or about 500,000 cells to about 1,000,000 cells. In some examples, the number of cells to be sorted is about 1 cell, about 100 cells, about 500 cells, about 1,000 cells, about 5,000 cells, about 10,000 cells, about 50,000 cells, about 100,000 cells, about 500,000 cells, or about 1,000,000 cells.

[0318] In some examples, the number of cells to be sorted is about 100 to about 500 cells, about 200 to about 500 cells, about 300 to about 500 cells, about 350 to about 500 cells, about 400 to about 500 cells, or about 450 to about 500 cells. In some examples, the reservoirs can be milliliter scale reservoirs. In some examples, the one or more reservoirs are pre-filled with a buffer and the sorted cells are stored in the buffer. Using the buffer helps to increase the volume of the cells, which can then be easily handled, for example a pipetted. In some examples, the buffer is a phosphate buffer, for example phosphate-buffered saline (PBS).

[0319] In some examples, the system and methods of the present disclosure comprise a cell sorting technique wherein pockets of buffer solution containing no negative objects are sent to the positive output channel in order to push rare objects out of the collection reservoir. In some examples, additional buffer solution is sent to the positive output channel to flush out all positive objects at the end of a run, once the channel is flushed clean (e.g., using the flush mechanism as disclosed herein).

[0320] In some examples, the system and methods of the present disclosure comprise a cell retrieving technique, wherein sorted cells can be retrieved for downstream analysis (e.g., molecular analysis). Non-limiting examples of the cell retrieving technique can include: retrieval by centrifugation; direct retrieval by pipetting; direct lysis of cells in well; sorting in a detachable tube; feeding into a single cell dispenser to be deposited into 96 or 384 well plates; etc.

[0321] In some examples, the system and methods of the present disclosure comprise a combination of techniques, wherein a graphics processing unit (GPU) and a digital signal processor (DSP) are used to run artificial intelligence (AI) algorithms and apply classification results in real-time to the system. In some examples, the system and methods of the present disclosure comprise a hybrid method for real-time cell sorting.

[0322] In some examples, the system and methods of the present disclosure comprise a feedback loop (e.g., an automatic feedback loop). For example, the system and methods can be configured to (i) monitor the vital signals and (ii) finetune one or more parameters of the system and methods based on the signals being read. At the beginning or throughout the run (e.g., the use of the microfluidic channel for cell imaging, classification, and/or sorting), a processor (e.g., a ML/AI

processor as disclosed herein) can specify target values for one or more selected parameters (e.g., flow rate, cell rate, etc.). In another example, other signals that reflect (e.g., automatically reflect) the quality of the run (e.g., the number of cells that are out of focus within the last 100 imaged cells) can be utilized in the feedback loop. The feedback loop can receive (e.g., in real-time) values of the parameters/signals disclosed herein and, based on the predetermined target values and/or one or more general mandates (e.g., the fewer the out-of-focus cells, the better), the feedback loop can facilitate adjustments (e.g., adjustments to pressure systems, illumination, stage, etc.). In some examples, the feedback loop can be designed to monitor and/or handle degenerate scenarios, in which the microfluidic system is not responsive or malfunctioning (e.g., outputting a value read that is out of range of acceptable reads).

[0323] In some examples, the system and methods of the present disclosure can adjust a cell classification threshold based on expected true positive rate for a sample type. The expected true positive rate can come from statistics gathered in one or more previous runs from the same or other patients with similar conditions. Such approach can help neutralize run-to-run variations (e.g., illumination, chip fabrication variation, etc.) that would impact imaging and hence any inference therefrom.

[0324] In some examples, the systems disclosed herein further comprise a validation unit that detects the presence of a particle without getting detailed information, such as imaging. In some instances, the validation unit can be used for one or more purposes. In some examples, the validation unit detects a particle approaching the bifurcation and enables precise sorting. In some examples, the validation unit detects a particle after the particle has been sorted to one of subchannels in fluid communication with the bifurcation. In some examples, the validation unit provides timing information with a plurality of laser spots, e.g., two laser spots. In some instances, the validation unit provides timing information by referencing the imaging time. In some instances, the validation unit provides precise time delay information and/or flow speed of particles.

[0325] In some examples, the particles (e.g., cells) analyzed by the systems and methods disclosed herein are comprised in a sample. The sample can be a biological sample obtained from a subject (e.g., a human or any animal). It should be appreciated that an animal can be a variety of any applicable type, including, but not limited thereto, mammal or non-mammals. For example, the animal can be veterinarian animal, livestock animal or pet type animal, etc. As an example, the animal can be a laboratory animal specifically selected to have certain characteristics similar to a human (e.g., rat, dog, pig, monkey, or the like). It should be appreciated that the subject can be any applicable human patient, for example. In some examples, the biological sample comprises a biopsy sample from a subject. In some examples, the biological sample comprises a tissue sample

from a subject. In some examples, the biological sample comprises liquid biopsy from a subject. In some examples, the biological sample can be a solid biological sample, e.g., a tumor sample. In some examples, a sample from a subject can comprise at least about 1%, at least about 5%, at least about 10%, at least about 15%, at least about 20%, at least about 25%, at least about 30%, at least about 35%, at least about 40%, at least about 45%, at least about 50%, at least about 55%, at least about 60%, at least about 65%, at least about 70%, at least about 75%, at least about 80%, at least about 85%, at least about 90%, at least about 95%, at least about 96%, at least about 97%, at least about 98%, at least about 99%, or at least about 100% tumor cells from a tumor.

[0326] In some examples, the sample can be a liquid biological sample. In some examples, the liquid biological sample can be a blood sample (e.g., whole blood, plasma, or serum). A whole blood sample can be subjected to separation of cellular components (e.g., plasma, serum) and cellular components by use of a Ficoll reagent. In some examples, the liquid biological sample can be a urine sample. In some examples, the liquid biological sample can be a perilymph sample. In some examples, the liquid biological sample can be a fecal sample. In some examples, the liquid biological sample can be saliva. In some examples, the liquid biological sample can be semen. In some examples, the liquid biological sample can be amniotic fluid. In some examples, the liquid biological sample can be cerebrospinal fluid. In some examples, the liquid biological sample can be bile. In some examples, the liquid biological sample can be sweat. In some examples, the liquid biological sample can be tears. In some examples, the liquid biological sample can be sputum. In some examples, the liquid biological sample can be synovial fluid. In some examples, the liquid biological sample can be vomit.

[0327] In some examples, samples can be collected over a period of time and the samples can be compared to each other or with a standard sample using the systems and methods disclosed herein. In some examples the standard sample is a comparable sample obtained from a different subject, for example a different subject that is known to be healthy or a different subject that is known to be unhealthy. Samples can be collected over regular time intervals, or can be collected intermittently over irregular time intervals.

Cell Feature Extraction: Output Data and Interpretation

[0328] FIG. 9 illustrates an example training architecture 900 for aspects herein disclosed systems, such as the human foundation model previously illustrated in FIG. 1. Architecture 900 can be completely symmetric, completely asymmetric, partially symmetric, or partially asymmetric, or a combination of the foregoing, and include a multi-layered (e.g., 18 layers deep) convolutional neural network. It is understood that after training, the system (e.g., system 100) can

be trained to predict morphometric characteristics of cellular images, including but not limited to cell class, cell type, cell state, other morphometric features such as blobs, related probabilities, and related accuracy identifiers.

[0329] In some examples, architecture 900 can be deployed on instruments (e.g., REM=I instrument of Table 4) and can be used to generate embeddings in a cloud-based computer system. Architecture 900 can be used in a high-throughput setting so that images 912. In some examples, images 912 are captured by a camera (e.g., an ultra high-speed bright-field camera) as cell suspensions flow through a channel in the microfluidics chip. Architecture 900 can include an augmentation module 940 configured to crop collected ultra-high-speed bright-field images 912 of cells as they pass through an imaging zone (e.g., an imaging zone of a microfluidic chip such as those captured images of FIGs. 8A-8B). Augmentation module 940 can implement one or more augmentation methods to generate batches 942a, b of altered replicas of the images 912. Augmentation techniques of module 940 includes, but is not limited to, horizontal and vertical flips of images, orthogonal rotation, translation, gaussian noise, contrast variation, and the like.

[0330] Batches 942a, b can be used to train a deep learning (DL) encoder 950. Specifically, batches 942a, b of altered replicas of the images 912 can be introduced along with images 912 into DL encoder 950 to generate augmented embeddings 952a, b. Encoder 950 can be trained using a self-supervised learning (SSL) method that learns image features without labels and relies at least on preserving information of its embeddings, including embeddings 952a, 952b, as well as concatenated deep learning predictive embeddings 964 (discussed more particularly below). DL encoder 950 can be a ResNet based encoder trained using a plurality of unlabeled cell images from different types of samples to detect differences in cell morphology without labeled training data. In some examples, encoder 950 learns image features without labels and with orthogonal morphometric features to improve model performance and interpretability.

[0331] Encoder 950 may include a plurality of convolution layers that use examples, such as edge detectors to detect a plurality of edge components of images 912 and batches 942a, b of altered replicas of the images 912. Encoder 950 can also use shape detectors to detect shape components of images 912 and batches 942a, b of altered replicas of the images 912 (e.g., a particular type of cell ridge). Augmented embeddings 952a, b from deep learning encoder 950 can be used to determine deep learning interpretations of captured images 912 performed in real-time (e.g., approximately <150 ms latency). To generate embeddings 952a, b, encoder 950 can encode features of batches 942a, b into multi-dimensional vectors. In some examples, encoder 950 can extract a 64-dimensional feature vector for each altered image of batches 942a, b and images 912.

[0332] Encoder 950 can be trained with a loss function that utilizes maximum likelihood-based

invariance between augmented images (such as mean squared error or categorical cross entropy), as well as variance, covariance, and morphometric decorrelation terms. In some examples, the variance and covariance terms used herein can include estimates of variance and covariance between feature dimensions by calculating the invariance and covariance directly for batches of images (e.g., of hundreds to thousands of images). In some examples, encoder 950 can be iteratively optimized until the DL model converges and calculate statistical quality (e.g., covariance) using the loss function. In some examples, encoder 950 can include a backbone, such as a ResNet-50 backbone, trained with the herein described invariance, variance, covariance, and morphometric decorrelation terms. The loss function uses an invariance term that learns invariance to vector transformations and is regularized with a variance term that prevents norm collapse. In some examples, the invariance term is determined using the mean square distance between embedding vectors (e.g., vectors of embeddings 952a, 952b). The loss function also uses a covariance term that prevents informational collapse by decorrelating the different dimensions of the vectors of embeddings 952a, 952b. The variance loss constrains the variance term of the vectors of embeddings 952a, 952b along each dimension independently. To determine similarity loss between vectors of embeddings 952a, 952b, a distance between vector pairs of embeddings 952a, b of the augmented images of batches 942a, b of the same cell is minimized (e.g., Euclidean distance) and variance of each embedding 952a, b over a training batch is maintained above a threshold. In some examples, the threshold is a hyperparameter determined by a value that gives us the best or most-optimized performance on downstream tasks. In some examples, variance is optimized to be around approximately 1. In some examples, variance can be optimized to be any value on a range of 0 or strictly (0, infinity] (strictly greater than 0).

[0333] With respect to the covariance term, different dimensions of embeddings 952a, 952b are used to make non-diagonal values of a cross-correlation matrix to zero indicating that the values are orthogonal. The covariances between embedding variables 952a, b over a respective batch between every pair of centered embedding variable is attracted to zero (e.g., see each of embeddings 952a, 952b) so that embeddings 952a, b are decorrelated from each other.

[0334] Architecture 900 can also include a computer vision encoder 960 that can be self-supervised and can include human-constructed algorithms, which in some cases can be referred to as the previously-described “rule-based morphometrics.” *See* Table 1. Encoder 960 may process captured images 912 as input and extract morphometric cell features into a plurality of morphometric vectors 962 (e.g., dimensional morphometric features encoded into 95-dimensional vectors representing the cell morphology). The multi-dimension vectors 962 can include cell position, cell shape, pixel intensity, pixel count, cell size, texture, focus, or combinations thereof.

In one nonlimiting example, encoder 960 can extract 99 dimensional embedding vectors representing cell morphology from high resolution images 912. In one nonlimiting example, previously described 64-dimensional embeddings 952a, b, and 51-dimensional morphometric features 962 can be encoded into 95-dimensional vectors representing the cell morphology.

[0335] Example depictions of certain contemplated morphometric cell features are shown in FIGs. 11A to 11B, where FIG. 11A illustrates representative images showing features that include cell shape and size (e.g., convex hull, max/min radius, max ferret diameter, min ferret diameter, long/short axis, etc.). FIG. 11B shows representative images showing features that include pixel intensity and texture (e.g., small white “blobs”, small black “blobs”, large white “blobs”, large white “blobs”, etc.). The illustrated morphometrics that describe size and intensities of “blobs” relate to cellular structures like granules, vesicles, and the like. In some examples, “blobs” can be understood as connected set(s) of pixels that are either substantially or entirely dark or substantially or entirely bright. In some examples, “blobs” can be understood region(s) in a respective image that differs in properties (e.g., brightness, color, etc.) relative to surrounding region(s).

[0336] In some examples, outputs of encoders 950, 960 can be analyzed together and concatenate as decorrelated concatenated morphometric predictive embeddings 964. Embeddings 964 can be generated adopting a probabilistic approach and/or using deep learning features of encoder 950 (e.g., using conditional batch normalization) concatenated with computer vision morphometric feature embeddings 962 from encoder 960 into different dimensions. Embeddings 964 can be predictive multidimensional vectors that include predictive features related to individual cells, clusters of cells, morphometric features, and related probabilities.

[0337] Embeddings 964 can be generated using morphometric decorrelation with encoder 950 minimizing the covariance term between vector pairs of embeddings 962 and rule-based morphometric dimensions over a training batch. Embeddings 964 can include encoded novel cell morphology information retained to be orthogonal to the rule-based morphometrics of encoder 960, including “blob” features such as those shown in FIGs. 11A to 11B. In so doing, architecture 900 effectively uses encoder 950 to splits its embedding layer into multiple sections to separate morphometric de-correlated embeddings 962 and morphometrics predictive embeddings 964 to predict or otherwise approximate morphometric features, such as “blobs”. In so splitting, architecture 900 is able to optimize conflict objectives in different parameter space and thus avoids adding latency to the system. In some examples, a correlation coefficient of predicted and/or approximated morphometric features, such as “blob” features, can be greater than approximately .9, though other ranges are contemplated as needed or required, including approximately .85 to .95.

[0338] In FIG. 10, architecture 900 is continued from FIG. 9 and shows that in one example a

multi-layer projector 970 can be included downstream of encoder 950. In this example, projector 970 is provided to project high dimension to low dimension (e.g., 2D), such as reducing the dimensionality of the embeddings 952a, 952b. In this respect, embeddings 952a, 952b are capable of being visualized. Projector 970 is configured to reduce the dimensionality to projected embeddings 972a, 972b and map representations of embeddings 952a, 952b. In this example, the previously discussed criterion, including the loss functions with invariance, variance, covariance, and morphometric decorrelation terms can be applied on projected embeddings 972a, 972b.

[0339] FIG. 12A illustrates cell classes, numbers of images used as training dataset to train a classifier using features extracted using the human foundation model, numbers of images processed by the human foundation model as test dataset, and corresponding representative cell images, in accordance with some examples of the present disclosure. A scale bar of 10 μm is shown on the representative cell images.

[0340] FIG. 12B illustrates a confusion matrix between predicted cell classes classified using features extracted using the human foundation model and actual cell classes, in accordance with some examples of the present disclosure. As illustrated, the human foundation model predicts cell classes with a high accuracy. The accuracy for predicting the Jurkat cell line, A375 cell line, and Caov-3 cell line is 90.5%, 89%, and 95.8%, respectively. The accuracy for predicting polystyrene beads is 100%.

Classifying and Sorting

[0341] Views (a) to (f) of FIG. 13 schematically illustrate an example system for classifying and sorting one or more cells. The platform as disclosed herein can allow for the input and flow of cells in suspension with confinement along a single lateral trajectory to obtain a narrow band of focus across the z-axis (views (a) to (f) of FIG. 13). View (a) of FIG. 13 shows the microfluidic chip and the inputs and output of the sorter platform according to one example of the present disclosure. Cells in suspension and sheath fluid are inputted, along with run parameters entered by the user: target cell type(s) and a cap on the number of cells to sort, if sorting is of interest. Upon run completion, the system generates reports of the sample composition (number and types of all of the processed cells) and the parameters of the run, including: length of run, number of analyzed cells, quality of imaging, quality of the sample. If sorting option is selected, it outputs isolated cells in a reservoir on the chip as well as a report of the number of sorted cells, purity of the collected cells and yield of the sort. Referring to view (b) of FIG. 13, a combination of hydrodynamic focusing and inertial focusing is used to focus the cells on a single z plane and a single lateral trajectory. Referring to views (c) and (d) of FIG. 13, the diagram shows the interplay between

different components of the software (view (c) of FIG. 13) and hardware pieces (view (d) of FIG. 13). The classifier is blown up in view (e) of FIG. 13, depicting the process of image collection, and automated real-time assessment of single cells in flow. After the images are taken, individual cell images are cropped using an automated object detection module, the cropped images are then run through a deep neural networks model trained on the relevant cells (e.g., DL encoder 950). For each image, the model can generate deep learning embeddings (e.g., embeddings 952a, 952b), deep learning predictive embeddings 964, as well as generate a prediction vector over the available cell classes and an inference will be made according to a selection rule (e.g., argmax). The model may also infer the z focusing plane of the image. The percentage of debris and cell clumps may also be predicted by the neural network model as a proxy for “sample quality”. View (f) of FIG. 13 shows the performance of sorting. In this figure, the tradeoff between purity and yield is shown in three different modes, for profiling as sorting of 130,000 [A], 500,000 [B] or 1,000,000 [C] cells within one hour.

[0342] Using a combination of hydrodynamic and inertial focusing, the platform can collect ultra high-speed bright-field images of cells as they pass through the imaging zone of the microfluidic chip (views (a) and (b) of FIG. 13). In order to capture the single cell images for processing, an automated object detection module can be incorporated to crop each image centered around the cell, before feeding the cropped images into a deep convolutional neural network (CNN) based on Inception architecture, which is trained on images of relevant cell types. In addition to classifying cells into categories of interest, the CNN can be trained to assess the focus of each image (in Z plane) and identify debris and cell clusters, thus providing information to assess sample quality (view (e) of FIG. 13). A feedback loop can be engineered so that the CNN inferred cell type can be used in real time to regulate pneumatic valves for sorting a cell into either the positive reservoir (cell collection reservoir) for a targeted category of interest or a waste outlet (FIG. 13A). Sorted cells in the reservoir may then be retrieved for downstream processing and molecular analysis. In some examples, the feedback loop can be engineered so that the generated deep learning embeddings (e.g., embeddings 952a, 952b, 964, etc.) can be used in real time to regulate pneumatic valves for sorting a cell into either a cell collection reservoir or a waste outlet (FIG. 13A).

[0343] FIG. 14 schematically illustrate operations that can be performed in an example method. View (a) of FIG. 14 shows high resolution images of single cells in flow are stored. Referring to view (b) of FIG. 14, AIAIA (AI Assisted Image Annotation) is used to cluster individual cell images into morphologically similar groups of cells. In some examples, AIAIA is used to cluster individual cell images into groups of cells using deep learning embeddings (e.g., embeddings 952a,

952b, 964). In some examples, a user uses the labeling tool to adjust and batch-label the cell clusters. In the example shown, one AML cell can be mis-clustered into a group of WBC cells and an image showing a cell clump (debris) can be mis-clustered in a NSCLC cell group. These errors are corrected by the “Expert clean-up” operation of view (b). Referring to view (c) of FIG. 14, the annotated cells are then integrated into a Cell Morphology Atlas (CMA). Referring to view (d) of FIG. 14, the CMA is used to generate both training and validation sets of the next generation of the models. Referring to view (e) of FIG. 14, during a sorting experiment, the pre-trained model shown in view (d) of FIG. 14 is used to infer the cell type (class) in real-time. The enriched cells are retrieved from the device. The retrieved cells are further processed for molecular profiling. The platform can be run in multiple different modes. In the training/validation mode, the collected images of a sample can be fed to the AIAIA, configured to use unsupervised learning to group cells into sub-clusters. In some examples, the sub-clusters be morphologically distinct sub-clusters. In some examples, the cells can be grouped using deep learning embeddings (e.g., embeddings 952a, 952b, 964). Using AIAIA, a user can clean up the sub-clusters by removing cells that are incorrectly clustered and annotates each cluster based on a predefined annotation schema. The annotated cell images are then integrated into the Cell Morphology Atlas (CMA), a growing database of expert-annotated images of single cells. The CMA is broken down into training and validation sets and is used to train and evaluate CNN models aimed at identifying cell types, cell states, morphometric features, and/or the like. Under the analysis mode (view (d) of FIG. 14), the collected images are fed into models that had been previously trained using the CMA, and a report is generated demonstrating the composition of the sample of interest. A UMAP visualization is used to depict the morphometric map of all the single cells within the sample. A set of prediction probabilities is also generated showing the classifier prediction of each individual cell within the sample belonging to every predefined cell class within the CMA. In the sorting mode (view (e) of FIG. 14), the collected images are passed to the CNN in real-time and a decision is made on the fly to assign each single cell to one of the predefined classes within the CMA. In some examples, the collected images are passed to the CNN in real-time and the decision is made on the fly to assign each single cell using deep learning embeddings (e.g., embeddings 952a, 952b, 964) within the CMA. The target cells are then sorted in real-time and are outputted for downstream molecular assessment.

[0344] FIG. 15 shows a computer system that is programmed or otherwise configured to implement methods provided herein. For example, the present disclosure provides computer systems that are programmed to implement methods of the disclosure. FIG. 15 shows a computer system 1501 that is programmed or otherwise configured to capture and/or analyze one or more

images of the cell. The computer system 1501 can regulate various examples of components of the cell sorting system of the present disclosure, such as, for example, the pump, the valve, and the imaging device. The computer system 1501 can be an electronic device of a user or a computer system that is remotely located with respect to the electronic device. The electronic device can be a mobile electronic device.

[0345] The computer system 1501 includes a central processing unit (CPU, also “processor” and “computer processor” herein) 1505, which can be a single core or multi core processor, or a plurality of processors for parallel processing. The computer system 1501 also includes memory or memory location 1510 (e.g., random-access memory, read-only memory, flash memory), electronic storage unit 1515 (e.g., hard disk), communication interface 1520 (e.g., network adapter) for communicating with one or more other systems, and peripheral devices 1525, such as cache, other memory, data storage and/or electronic display adapters. The memory 1510, storage unit 1515, interface 1520 and peripheral devices 1525 are in communication with the CPU 1505 through a communication bus (solid lines), such as a motherboard. The storage unit 1515 can be a data storage unit (or data repository) for storing data. The computer system 1501 can be operatively coupled to a computer network (“network”) 1530 with the aid of the communication interface 1520. The network 1530 can be the Internet, an internet and/or extranet, or an intranet and/or extranet that is in communication with the Internet. The network 1530 in some cases is a telecommunication and/or data network. The network 1530 can include one or more computer servers, which can enable distributed computing, such as cloud computing. The network 1530, in some cases with the aid of the computer system 1501, can implement a peer-to-peer network, which may enable devices coupled to the computer system 1501 to behave as a client or a server.

[0346] The CPU 1505 can execute a sequence of machine-readable instructions, which can be embodied in a program or software. The instructions can be stored in a memory location, such as the memory 1510. The instructions can be directed to the CPU 1505, which can subsequently program or otherwise configure the CPU 1505 to implement methods of the present disclosure. Examples of operations performed by the CPU 1505 can include fetch, decode, execute, and writeback.

[0347] The CPU 1505 can be part of a circuit, such as an integrated circuit. One or more other components of the system 1501 can be included in the circuit. In some cases, the circuit is an application specific integrated circuit (ASIC).

[0348] The storage unit 1515 can store files, such as drivers, libraries and saved programs. The storage unit 1515 can store user data, e.g., user preferences and user programs. The computer system 1501 in some cases can include one or more additional data storage units that are external

to the computer system 1501, such as located on a remote server that is in communication with the computer system 1501 through an intranet or the Internet.

[0349] The computer system 1501 can communicate with one or more remote computer systems through the network 1530. For instance, the computer system 1501 can communicate with a remote computer system of a user. Examples of remote computer systems include personal computers (e.g., portable PC), slate or tablets, telephones, Smart phones (e.g., Apple® iPhone, Android-enabled device, Blackberry®), or personal digital assistants. The user can access the computer system 1501 using the network 1530.

[0350] Methods as described herein can be implemented by way of machine (e.g., computer processor) executable code stored on an electronic storage location of the computer system 1501, such as, for example, on the memory 1510 or electronic storage unit 1515. The machine executable or machine readable code can be provided in the form of software. During use, the code can be executed by the processor 1505. In some cases, the code can be retrieved from the storage unit 1515 and stored on the memory 1510 for ready access by the processor 1505. In some situations, the electronic storage unit 1515 can be precluded, and machine-executable instructions are stored on memory 1510.

[0351] The code can be pre-compiled and configured for use with a machine having a processor adapted to execute the code, or can be compiled during runtime. The code can be supplied in a programming language that can be selected to enable the code to execute in a pre-compiled or as-compiled fashion.

[0352] Examples of the systems and methods provided herein, such as the computer system 1501, can be embodied in programming. Various examples of the technology can be thought of as “products” or “articles of manufacture” typically in the form of machine (or processor) executable code and/or associated data that is carried on or embodied in a type of machine readable medium. Machine-executable code can be stored on an electronic storage unit, such as memory (e.g., read-only memory, random-access memory, flash memory) or a hard disk.

[0353] “Storage” type media can include any or all of the tangible memory of the computers, processors or the like, or associated modules thereof, such as various semiconductor memories, tape drives, disk drives and the like, which may provide non-transitory storage at any time for the software programming. All or portions of the software may at times be communicated through the Internet or various other telecommunication networks. Such communications, for example, may enable loading of the software from one computer or processor into another, for example, from a management server or host computer into the computer platform of an application server. Thus, another type of media that may bear the software elements includes optical, electrical and

electromagnetic waves, such as used across physical interfaces between local devices, through wired and optical landline networks and over various air-links. The physical elements that carry such waves, such as wired or wireless links, optical links or the like, also can be considered as media bearing the software. As used herein, unless restricted to non-transitory, tangible “storage” media, terms such as computer or machine “readable medium” refer to any medium that participates in providing instructions to a processor for execution.

[0354] Hence, a machine readable medium, such as computer-executable code, may take many forms, including but not limited to, a tangible storage medium, a carrier wave medium or physical transmission medium. Non-volatile storage media include, for example, optical or magnetic disks, such as any of the storage devices in any computer(s) or the like, such as can be used to implement the databases, etc. shown in the drawings. Volatile storage media include dynamic memory, such as main memory of such a computer platform. Tangible transmission media include coaxial cables; copper wire and fiber optics, including the wires that comprise a bus within a computer system. Carrier-wave transmission media may take the form of electric or electromagnetic signals, or acoustic or light waves such as those generated during radio frequency (RF) and infrared (IR) data communications. Common forms of computer-readable media therefore include for example: a floppy disk, a flexible disk, hard disk, magnetic tape, any other magnetic medium, a CD-ROM, DVD or DVD-ROM, any other optical medium, punch cards paper tape, any other physical storage medium with patterns of holes, a RAM, a ROM, a PROM and EPROM, a FLASH-EPROM, any other memory chip or cartridge, a carrier wave transporting data or instructions, cables or links transporting such a carrier wave, or any other medium from which a computer may read programming code and/or data. Many of these forms of computer readable media can be involved in carrying one or more sequences of one or more instructions to a processor for execution.

[0355] The computer system 1501 can include or be in communication with an electronic display 1535 that comprises a user interface (UI) 1540 for providing, for example, the one or more images of the cell that is transported through the channel of the cell sorting system. In some cases, the computer system 1501 can be configured to provide a live feedback of the images. Examples of UIs include, without limitation, a graphical user interface (GUI) and web-based user interface.

[0356] Methods and systems of the present disclosure can be implemented by way of one or more algorithms. An algorithm can be implemented by way of software upon execution by the central processing unit 1505. The algorithm can include, for example, the human foundation model.

[0357] It will be appreciated that the features and operations described herein can be used in any suitable combination with one another. For example, FIG. 16 illustrates an example flow of operations in a method of processing images. Method 1600 illustrated in FIG. 16 includes

extracting, using a Deep Learning (DL) model (e.g., DL encoder 950), a set of machine learning (ML)-based features from a cell image (e.g., images 912 and/or augmented images thereof such as in batches 942a, 942b) (operation 1610). As one example, any suitable component(s) may include a processor and a non-computer readable medium storing the machine learning model and related encoder, and instructions for causing the processor to perform operations. For example, the processor can be included with a cloud-based computing environment and/or within a microfluidics platform (e.g., platform 20, 310). Nonlimiting examples of machine learning encoders (such as deep learning encoders, for example, convolutional neural networks) are provided elsewhere herein. In some examples, cells in the one or more cell images are unstained. The one or more cell images are brightfield cell images in some examples, though it will be appreciated that other types of cell images readily can be provided to the machine learning encoder and computer vision encoder.

[0358] Method 1600 illustrated in FIG. 16 also may include generating, using the DL model and the set of ML-based features, a plurality of DL embeddings orthogonal to each other (operation 1620).

[0359] Method 1600 illustrated in FIG. 16 also may include extracting, using a computer vision model, a set of cell morphometric features from the cell image, the cell morphometric features being orthogonal to the plurality of DL embeddings (operation 1630). Some nonlimiting examples of cell morphometric features include cell position, cell shape, pixel intensity, texture, focus, or any combination thereof. These and other nonlimiting examples of cell morphological features are described in Table 2.

[0360] Method 1600 illustrated in FIG. 16 also may include generating a plurality of morphometric predictive embeddings using the set of ML-based features and the set of cell morphometric features (operation 1640).

[0361] FIG. 17 illustrates an example flow of operations in a method of processing images. Method 1700 illustrated in FIG. 17 includes extracting, using a Deep Learning (DL) model (e.g., DL encoder 950) and image data of a plurality of cells, a vector for a cell of the plurality of cells, the vector comprising a set of machine-learning (ML)-based features and a set of cell morphometric features extracted using a computer vision model (operation 1710). As one example, any suitable component(s) may include a processor and a non-computer readable medium storing the machine learning model and related encoder, and instructions for causing the processor to perform operations. For example, the processor can be included with a cloud-based computing environment and/or within a microfluidics platform (e.g., platform 20, 310). Nonlimiting examples of machine learning encoders (such as deep learning encoders, for example, convolutional neural

networks) are provided elsewhere herein. In some examples, cells in the one or more cell images are unstained. The one or more cell images are brightfield cell images in some examples, though it will be appreciated that other types of cell images readily can be provided to the machine learning encoder and computer vision encoder.

[0362] Method 1700 illustrated in FIG. 17 also may include generating, using the DL model and using the set of ML-based features, a plurality of DL embeddings orthogonal to each other (operation 1720).

Feature Space for Cell Images

[0363] Cell images may be represented as points in a space (referred to herein as “feature space”) defined by the information included in the image. To illustrate, consider a black and white cell image encoded as an 256×256 array of pixels, each of which has a value ranging from 0 (for black pixels) to 255 (for white pixels). One way for such an image to be represented as a location feature space may be to use a feature space with one dimension (i.e., a coordinate determining a position in the feature space, just as length, width and height determine a physical position in three dimensional physical space), for each pixel (which, in this example, may be $256 * 256 = 65,536$ dimensions). In this this way, each cell image (and, indeed, any image) encoded in the specified format may be represented as a unique location in the feature being considered.

[0364] In one example, while representing cell images as being located in a feature space, which directly translates pixels into dimensions, is possible, in this example this does not provide anything beyond the image itself. However, it is possible for representing an image as a location in space to provide advantages over the image itself. An example which can illustrate this is representing images black circles which are encoded in the same 256×256 , black and white format described above can as locations in feature space. In this case, rather than using a feature space with one dimension for each pixel, the images can be represented as locations in a feature space having only three dimensions – one dimension for the x coordinate of the circle’s center, one dimension for the y coordinate of the circle’s center, and one dimension for the circle’s radius. This type of representation may have significant advantages over the original image. For example, rather than being stored using 65,536 values – i.e., one value for each pixel – the location of a black and white circle image in this feature space may be stored using only three values – i.e., one for each dimension.

[0365] The approach described above – i.e., representing images as locations in a feature space whose dimensions are based on the information content of the images to be so represented – can also be applied to images which are more complicated than images of geometric primitives. However, for these more complicated types of images, such as cell images, determining

appropriate dimensions for the feature space in which the images may be located may not be straightforward. As a result, various tools may be used to identify those dimensions, and to generate an embedding (i.e., a vector having one value for each of the dimensions in the applicable feature space) from an input image. For instance, in some cases, dimensions may be identified through use of an encoder comprising a neural network with a plurality of convolution layers, which may identify features of an input image and generate a reduced dimensionality output image, and a plurality of transpose convolution layers, which may generate an increased dimensionality output image based on features included in an input image. Illustrations of potential operations which may be performed by the layers of such a network are discussed below in the context of FIGS. 18 and 19.

[0366] As shown in FIG. 18, a convolution layer may identify features in an input image 101, and generate a reduced dimensionality output image 102 by convolving the input image 101 with a convolution filter 103. Specifically, FIG. 18 illustrates how convolution of square input image with $n+1$ pixel long sides may be used to generate a square output image with $n-1$ pixel long sides through convolving the square input image with a 3×3 convolution filter. To implement this type of operation in a neural network, the input and output images may each be represented by layers in the network. The weights on the connections between those layers may then define the convolution operation. For example, given a 3×3 convolution filter such as shown in FIG. 18, a pixel at location (a,b) in the output image may have a weight for the pixel at location $(a-1,b-1)$ of the input image equal to the top left value in the convolution filter, a weight for the pixel at location $(a,b-1)$ of the input image equal to the top center value in the convolution filter, and may continue in this pattern for the remainder of the square extending from $(a-1,b-1)$ to $(a+1,b+1)$ in the input image, while having weight values of zero elsewhere. The particular values in the filter may then define the feature of the input image which may be captured in a convolution layer's output image. For example, a 3×3 filter with a value of 8 at its center and -1 elsewhere such as shown in Table 3, below, may capture the edges from the input image.

[	-1	-1	-1	]
[	-1	8	-1	]
[	-1	-1	-1	]

Table 3

[0367] Multiple such convolution layers may be arranged in a series (i.e., the output image of one layer may serve as the input image to the next) to further refine the features and reduce the dimensionality of the ultimate output image created by the convolution layers.

[0368] As shown in FIG. 19, a transpose convolution layer may function in a manner similar to

that described above for a convolution layer, with a filter 201 being used to create an output image 202 from an input image 203. However, in one example, so that the dimensionality of the output image 202 is greater than that of the input image 203, the input image 203 may be augmented by adding zeros around its periphery and between its elements, with the filter 201 only being applied to the expanded image 204 obtained using this augmentation. By chaining together transpose convolution layers which operate in a manner such as illustrated in FIG. 19, with the first transpose convolution layer taking the output 102 of the last convolution layer as input 203, it is possible to obtain a final output image which has the same resolution as the original input image to the first convolution layer. This final output image can then be compared with the original image provided as input to the first convolution layer, and used with a loss function (e.g., mean squared error, binary cross entropy) to allow a neural network comprising convolution and transpose convolution layers to be trained using unlabeled images of the type to be represented (e.g., cell images). The pixels in the smallest output image (i.e., the output image created by the final convolution layer which is provided as input to the first transpose convolution layer) can then be treated as dimensions in the feature space where the images may be located. Similarly, the convolution layers can be used as an encoder by treating the values of the pixels in the smallest output image as the embedding for that image.

[0369] Other approaches to defining a space for representing complex image types are also possible, and may be used in some implementations of the disclosed technology. For example, rather than using machine learning to identify dimensions of a space for representing images, it is possible that physical features of objects depicted in the images may be treated as spatial dimensions, and that those features may be identified using dedicated computer vision functions to generate embeddings for particular images. As an illustration of this, consider FIGS. 20A-20E, which illustrate morphometric features which may be extracted from cell images and used as dimensions in a feature space, either in addition to, or as alternatives to, dimensions derived using machine learning such as described above. Further approaches to identifying dimensions for feature spaces in which images can be represented, as well as for generating embeddings for representing images in those spaces, are also possible, and may be utilized in various applications of the disclosed technology. For example, principal component analysis, as well as approaches such as described in Salek, et al., Realtime morphological characterization and sorting of unlabeled viable cells using deep learning, available at <https://www.biorxiv.org/content/10.1101/2022.02.28.482368v2> or Bank, et. al., Autoencoders, available at <https://arxiv.org/pdf/2003.05991.pdf>, the disclosure of each of which is hereby incorporated by reference in its entirety, may also be applied in some cases.

[0370] While the above description of feature spaces for cell images has provided various examples, such examples are intended to be illustrative only, not to imply limitations. For instance, FIGS. 18 and 19 illustrated various items as having particular dimensions (e.g., 3x3 convolution filter in FIG. 1; 3x3 transpose convolution filter, 3x3 input image, and 5x5 output image in FIG. 19). However, those dimensions are used only for illustration, and other dimensions may also be used. For example, convolution and transpose convolution filters having dimensions other than 3x3 may be used, and transpose convolution layers may have input images with sizes other than 3x3 and output images with sizes other than 5x5. Similarly, while the output images of FIGS. 18 and 19 may be obtained by applying filters on a pixel by pixel basis (e.g., a filter may be convolved with one 3x3 area, then moved one pixel and convolved with another 3x3 area, etc.) it is also possible that filters may be applied on a different basis in some scenarios (e.g., a filter may be convolved with one 3x3 area, then moved two pixels and convolved with another 3x3 area, etc.). Similarly, the amount of zeros added between pixels in, and/or around the periphery of, an input image to create an enhanced image 204 may be more than the single rows and columns of zeros illustrated in FIG. 19.

[0371] It is also possible that additional operations beyond convolving and adding zeros may be used to increase or reduce the size of images when a machine learning approach is applied. For example, pooling, downsampling, upsampling, or a combination thereof, may be used to increase or decrease the size of images as appropriate in some cases. Indeed, in some cases approaches such as pooling, downsampling or upsampling may be the only operations which may be used to reduce size. For instance, in some cases a convolution layer may include augmenting an input image by adding zero padding so that the output provided by applying a convolution filter may have the same dimension as the input image, and then the size of that output image may be reduced by application of a pooling operation. Other suitable approaches may be employed. Accordingly, the above variations, like the discussions of FIGS. 18 and 19 themselves, are to be illustrative only, and not limiting.

[0372] In an example to further illustrate potential variations on the approaches described above, the loss function may be used when using neural networks to identify the dimensions of a feature space in which cell images may be represented. While, as noted above, the use of convolution and transpose convolution layers can allow such a neural network to be trained on unlabeled data based on how well it can reconstruct a cell image, in some cases labeled data may also be used in the training process. For example, a loss function may be implemented to combine the loss based on how well the neural network can reconstruct a cell image (referred to as “reconstruction loss”), with how well the dimensions in a particular feature space function in

allowing a cell image to be classified. For instance, the values of pixels at the feature space dimensions may be used as inputs to a dense network with output nodes corresponding to labels of cell images with ground truth (e.g., provided by a human annotator) classifications, and the loss from comparing the ground truth classification with the classifications provided by the dense network may be combined with the reconstruction loss when training the convolution layers of the network. Also, while the example of representing a circle using a feature space defined by the circle's radius and the coordinates of its center allowed the circle's location in feature space to completely capture all information from the original image, not all embeddings will provide such a lossless encoding. Accordingly, the above descriptions of how dimensions of an embedding space for images such as cell images, as well as the analogy of such embedding spaces to other embedding spaces such as geometric embedding spaces, is to be illustrative only, and not limiting.

Generating Cell Images

[0373] As set forth above it is possible to generate a cell image by providing an embedding to a deep learning model (e.g., a series of transpose convolution layers such as described in the context of FIG. 19) designed to translate embeddings into images (such a series of layers referred to herein as a “decoder”). However, other approaches for generating cell images are also possible. For example, a neural network can be trained to generate cell images from random input images (e.g., a picture of static where the values of the pixels in the picture follow a gaussian distribution) by, for each of a set of training images, progressively adding more noise to that image until an obscured image of effectively random pixels is produced, and training the network to approximate the (actually intractable) process of recreating the original input. A method which may be used to perform this type of training is illustrated in FIG. 21, discussed below.

[0374] As shown in FIG. 21, a method for training a neural network to approximate the process of denoising cell images can start in block 401 with obtaining an input image. This may be done, for example, by retrieving a cell image from a set of training data and passing it to a function which may perform a training process such as shown. After the input image is obtained in block 401, an output image can be obtained by adding noise (e.g., gaussian white noise) to the input image in block 402. This can be done, for example, by using an equation such as equation 1, below in which x_t is the output image and x_{t-1} is the input image, to progressively add more noise to the image on each iteration of the process shown in FIG. 21 by increasing β from a relatively low value initially (e.g., 0.00085) to a relatively higher value (e.g., 0.012) as the image is more completely obscured.

$$x_t = \sqrt{1 - \beta_t} x_{t-1} + \sqrt{\beta_t} \epsilon$$

Equation 1

[0375] After this, the neural network can be trained to generate the input image from the output image (which, in this example, may be done by training the neural network to approximately denoise the output image) in block 403. This can be done in a manner similar to that described in the context of FIGS. 18 and 19 for training a decoder to generate an output image based on an embedding. However, in this case, rather than being provided with an embedding (as was the case with the decoder discussed previously), the neural network may be provided with the output image and its loss may be based on the neural network's ability to generate a new image that matched the input image as it appeared before noise was added in block 402. After the training had taken place, a decision may be made in block 404 of whether to perform further iterations of the process of FIG. 21. This decision may be implemented by, for example, checking if a predefined number of iterations (e.g., between 20 and 25 iterations) had already taken place. Once the decision of block 404 is made, if it was to continue iterating, then blocks 401 may be repeated, with the output image obtained on the previous iteration being treated as the input image for the further iteration of blocks 401-403. Otherwise, if the decision in block 404 is that no more iterations are needed, the process for FIG. 21 may be treated as done in block 405, and it may then be repeated for the next image in the training data (e.g., an additional cell image from a database of cell images).

[0376] For a neural network trained in a manner such as that shown in FIG. 21 to create new images, a process such as shown in FIG. 22 may be used. In that process, in block 501, an input may be received. This may be done, for example, in an initial iteration of the process of FIG. 22 by providing a random input to a neural network trained in a manner such as described above in the context of FIG. 21. In subsequent iterations, obtaining the input in block 501 may be performed by feeding an output obtained on a previous iteration back into the network as its new input data. The input (whether on the initial iteration or otherwise) may then be de-noised in block 502 to provide a de-noising output. This may be done by actually applying the neural network trained as described in the context of FIG. 21 to the input obtained in block 501. In block 503, a decision may be made as to whether further iterations are needed. This may be done, for example, by continuing to iterate the process of FIG. 22 for the same number of iterations as were used in training the neural network using the process of FIG. 21 (e.g., in implementations in which there was a fixed number of iterations per image in the training loop of FIG. 21). In some examples, other approaches to making the decision in block 503 may be used. For example, in some cases the number of de-noising iterations may be a user definable parameter, and so the individual using the network to create a new image may provide the information for deciding if more iteration are needed. If more

iterations are needed, then the process may repeat, with the output of the previous iteration being treated as the input for the next iteration. Otherwise, the process of FIG. 22 may terminate in block 504, with the final de-noising output (e.g., a new picture created based on the initial random input) treated as the ultimate output of the neural network.

[0377] Variations on the above described approaches for training and using a neural network to create new images are also possible, and may be implemented by one of skill in the art without undue experimentation in light of this disclosure. For example, in some cases, rather than progressively adding noise to an image and training a neural network to recreate the input image by removing added noise, noise may be progressively added to an embedding for an image, and the neural network may be trained to recreate the embedding by removing the added noise. In this case, rather than the neural network creating a new image directly, it may be used to create an embedding, and a decoder may then create the new image from the embedding. As another example, in some cases statistical methods, such as comparing an image with Gaussian noise using tests such as D'Agostino-Pearson, Anderson-Darling, Shapiro-Wilk and Kolmogorov-Smirnov may be used to control the additional of noise (e.g., in blocks 402 and/or 404 of FIG. 21). As another example, in some cases, rather than providing random inputs, a trained neural network may be provided with additional information, such as a complete or partial embedding, and this additional information can be used to guide the image creation process. FIG. 23, discussed below, illustrates in one example a process in which the creation of a new image can be guided by an embedding or partial embedding (such embedding or partial embedding referred to as a “prompt” in the context of FIG. 23).

[0378] As shown in FIG. 23, the process of guiding creation of a new image using a prompt may be largely similar to the de-noising process of FIG. 22. In the process of FIG. 23, a prompt may be obtained in block 601. This may be done, for example, by a user specifying values for certain dimensions in feature space (e.g., values for roundness, ellipse elongation and convex shape, in a case where the feature space dimensions include morphometric parameters such as shown in FIG. 20B), though other ways are also possible and are discussed in more detail in section IV. After the prompt is obtained, an attention output may be created from that prompt in block 602. This may be done, for example, by projecting the feature space dimensions with values defined in the prompt onto a flattened representation of the neural network using a set of trainable projection matrices. This attention output may then be added to the de-noising output (i.e., the result of applying de-noising in block 502, as described in the context of FIG. 22) to create a conditioned output in block 603. This conditioned output may then be either treated as the final output of the neural network (if the decision in block 503 indicates that no further iterations are needed), or may

be treated as input and itself be subjected to denoising (if the decision in block 503 indicates that further iterations are to take place) in the process of obtaining the final output of the network. This final output (i.e., an embedding created by progressive denoising as illustrated in FIG. 23) may then be provided to a decoder, which may ultimately use it to generate an output image.

[0379] Other variations are also possible, both in training a machine learning model to create new images and in actually using a trained model in image creation. For example, approaches such as described in Mirza et. al., Conditional Generative Adversarial Nets, available at <https://arxiv.org/abs/1411.1784>, Rombach, et. al., High-Resolution Image Synthesis with Latent Diffusion Models, available at <https://arxiv.org/abs/2112.10752>, Esser, et. al., Taming Transformers for High-Resolution Image Synthesis, available at <https://arxiv.org/abs/2012.09841>, Alvaro Figueira and Brunon Vaz, Survey on Synthetic Data Generation, Evaluation Methods and GANs, available at <https://www.mdpi.com/2227-7390/10/15/2733>, the disclosures of which are each incorporated herein by reference in their entirety, may also be used. Accordingly, the above discussion of cell image generation is to be understood as being illustrative only, and not limiting

User Conditioned Image Generation

[0380] As set forth above trained neural networks can be used to create cell images based on prompts defining values for dimensions of the cell images in feature space. This may allow a user to guide the image creation process for a variety of potential end uses. FIG. 24 depicts in one example a process in which the disclosed technology can be used in exploring the transition between different cell states, such as the transition between melanocytic and mesenchymal melanoma cells. Initially, in the process of FIG. 24, first and second states may be defined in block 701. This may be done in a variety of manners. For example, in some cases, a user may provide two cell images as input, and the first cell image may be treated as the first state, and the second cell image may be treated as defining the second state. In some other examples, a user may specify cell states by name, such as using a drop down or text entry field allowing him or her to provide the names of the relevant states. In such a case, a computer which is being used to perform the method of FIG. 24 may compare the user's entered text with a database of cell states, and then treat the cell states which most closely matched the user's input as the first and second states for purposes of the process of FIG. 24. Other acts (e.g., informing the user of the closest matches from the database of cell states and asking them to confirm that those states are really the states they intended) are also possible, and may be included in some implementations following the diagram of FIG. 24.

[0381] After first and second states (e.g., melanocytic and mesenchymal melanoma cells) had

been defined in block 701, embeddings may be determined for each of those states in block 702. As with the original definition of first and second states, the embedding definition of block 702 may be performed in a variety of manners. For example, in a case where the first and second states were defined by a user providing images for the first and second states, the embeddings for the first and second states may be determined by providing the user's images as inputs to an encoder trained to generate embeddings for cell images. In another example, in a case where first and second states were defined by matching user specified state names against a database of cell states, each cell state in the database may have a predefined archetypal embedding, and so determining the embeddings for the first and second states in block 702 may be performed by retrieving the applicable already defined embeddings. Other approaches are also possible, and may be implemented without undue experimentation based on this disclosure. Accordingly, the above description of how embeddings may be determined for the first and second states in a method such as shown in FIG. 24 is to be understood as being illustrative only, and is not to be treated as limiting.

[0382] Once the first and second state embeddings had been determined in block 702, those embeddings may be used to determine a set of transitional dimensions in block 703. This determination may be performed by identifying how the embeddings for the first and second state differed from each other, and treating the dimensions whose values in the embeddings differed most as the transitional dimensions. The process of FIG. 24 may then proceed to a loop in which a set of prompts may be created by modifying the values of those transitional dimensions to illustrate the transition between the first and second (or, in some versions, the transitional dimensions form the first embedding) as a current prompt. With the current prompt set, an additional prompt may be created by modifying the current prompt in block 705. For example, in a case where the process of FIG. 24 is being performed by a system which may present a predefined number of transitional images, the differences between the first and second states for each of the transitional dimensions may be divided by the predefined number of transitional images, and the current prompt may be modified in block 705 by adding the result of that division to each of the transitional dimensions. As another example of how this modification may take place, in some scenarios, the dimensions in the latest space for cell images may be associated with modification increments (e.g., minimum changes for those dimensions which may be likely to be detectable to a human viewer) and those modification increments may be used to modify the current prompt in block 705. For instance, this may be done by: (1) identifying the dimension where the ratio of the difference between the embeddings for the first and second states to that dimension's modification increment may be identified, (2) defining a number of stages as equal to the number of times that dimension's modification increment goes into the difference between its value in the embeddings for the first

and second states; and (3) modifying each of the transitional dimensions by adding the result of dividing the difference between that dimension's value in the embeddings for the first and second states by the number of stages.

[0383] Whatever modifications are performed, once the additional prompt has been obtained, a determination may be made in block 706 of whether the second state had been reached. This may be done, for example, by comparing the values of the additional prompt against an embedding for the second state and, if they matched (either exactly or within tolerance thresholds that had previously been defined for the various dimensions), then the determination may be that the second state had been reached. In another example, in some cases, if a process such as shown in FIG. 24 is performed by a system which may create a defined (either predefined or defined dynamically, such as via modification increments) number of additional prompts, the determination of block 706 may be made by checking if the defined number of additional prompts had, in fact, been created. If it had not, then the most recently created additional prompt may be set as the current prompt in block 707, and the modification look may continue. Otherwise, images may be generated in block 708 for the first state's embedding and each additional prompt, such as by providing the additional prompt to a decoder to use in creating the images, by using the additional prompt to condition the output of a machine learning model trained to approximate a de-noising process such as described in the context of FIG. 23, or by providing only the transitional dimensions to a machine learning model trained to perform such conditional de-noising. However they are generated, once the images are available, they may, in block 709, be displayed to a user, such as via a user interface generated by a system which performed the method of FIG. 24. This interface may display the images showing the transition from the first state to the second state, and may also include other information which may be of benefit to a user. To illustrate, consider FIG. 25, discussed below, which depicts an interface which may be used to display images such as those which may be generated in block 708.

[0384] FIG. 5 illustrates a set of synthetic cell images which may be created using the described technology to illustrate how a melanoma cell may change when transitioning from the melanocytic to the mesenchymal state. Additionally, the interface of FIG. 25 also shows particular micro-events which take place during the transition, which micro-events may or may not correspond to individual dimensions in the latent space. For instance, as shown in FIG. 25, an interface that may be used to display generated images in block 709 may include columns showing membrane changes, elongation, dark circumferential features, and cytoplasmic features. To generate this type of interface, a system implemented based on FIG. 24 may be programmed with a set of visual characteristics, as well as how those characteristics may be determined from cell images (e.g.,

membrane changes may be identified based on changes in circularity of a cell image; dark circumferential features may be identified based on relatively dark portions around the perimeter of a cell image), and how those characteristics may be illustrated for a user (e.g., membrane changes may be illustrated through highlighting the perimeter of a cell; elongation may be illustrated using a line along the major axis of a cell). Then, when creating an interface such as shown in FIG. 25, the visual characteristics which change most significantly from the first to the second states may be identified and used to populate the interface for review by the user.

[0385] In addition to allowing the tracking of micro-events such as described above in the context of FIG. 25, other uses of processes such as shown in FIG. 24 for the generation of synthetic sequences are also possible, and may be practiced based on this disclosure. For example, comparing what appears to be shown in synthetic images with what is known in the literature may be used for hypothesis generation to allow a user to more effectively identify research paths which may help improve his or her (or the field in general's) understanding of the transition between the first and second states. For example, the illustration in synthetic images created using a process such as shown in FIG. 24 that cell membranes become more irregular in the mesenchymal state may drive a hypothesis such as that migratory behavior is allowed in the mesenchymal state as a result of the cell membrane becoming more fluid in the transition from the melanocytic state. This type of hypothesis may be based on a user's individual observation and knowledge of the literature, but may also be performed automatically or semi-automatically, such as based on use of embeddings similar to those described above with respect to representing images as locations in space. For instance, in some cases, changes which may be illustrated in a series of synthetic images, may be associated with various descriptive terms (e.g., membrane changes represented by decreased circularity may be associated with descriptors like irregular) and semantic dimensions for those changes may be identified by creating embeddings for the descriptive terms, as well as other terms associated with the images like mesenchymal (e.g., using software such as Word2Vec, available from Google, Inc. at <https://code.google.com/archive/p/word2vec/>). Those semantic dimensions may then be compared with subjects addressed in the literature, and references having the greatest overlap may be identified as suggesting potentially fruitful paths for future research based on the generated images.

[0386] The disclosed technology may also be used to support insights into natural phenomenon which may not be possible purely from the observation of natural cells. To illustrate, consider the example above of depicting the transition between melanocytic and mesenchymal melanoma cells. While it may not be practical to obtain images of cells at arbitrary points on the transition between these states, this is at least theoretically possible. By contrast, obtaining an image depicting a cell

in a 110% mesenchymal state is not something that may be done with real cells, even theoretically. However, with the disclosed technology images of a cell at a 110% mesenchymal state (or 120%, or any other percent) may be done by continuing the modification loop shown in FIG. 24 until 110% mesenchymal (or other desired state) had been reached, thereby potentially providing insights into what the state of “mesenchymal-ness” actually means. Similarly, using the disclosed technology, it can be possible to observe transitions which do not happen in nature. For example, by using cell images marked as “viable” and “non-viable,” it can be possible to view what a transition from a non-viable to a viable cell may look like, potentially even extending this transition to a cell which was more than 100% viable (or less than zero percent nonviable, if the transition was extended in the other direction), thereby potentially supporting insights into concepts like viability or non-viability.

[0387] Another example of a potential use for synthetic image generation technology such as described herein is to increase the interpretability of otherwise opaque decisions which may be made by an artificial intelligence system. To illustrate, consider an artificial intelligence system which had been trained to classify cell images by creating representations of those images in a feature space and then providing the values of the dimensions in a cell image’s feature space representation to a dense network as inputs. Regardless of their accuracy, the reasons for those classifications may not be readily understandable by human beings, since feature space dimensions identified by a machine learning model may not correspond to commonly recognized visual attributes such as circularity or elongation. To help address this, a system may be implemented which allows a user to manipulate one or more feature space dimensions of interest (e.g., feature space dimensions which led to a classification that the user may like to better understand), and the disclosed technology may be used to generate cell images showing the impact of those manipulations. Additionally, in some cases these generated cell images may be presented in a manner to make the impact of changes in particular feature space dimensions clearer. For example, a system implemented based on this disclosure may display a synthetic cell image with changes highlighted (e.g., by subtracting an image generated with the user’s modifications from a baseline image where those modifications were not present, and then presenting the pixels which differed between the two images in a contrasting color), thereby allowing the user to obtain a more intuitive visual understanding of the feature space dimensions used by the machine learning model.

[0388] The disclosed technology may also be applied to allow exploration and/or elaboration of the feature space in which cell images are represented. For example, in some cases, a user may be able to define a vector, such as by defining a trend a cell might undergo (e.g., becoming cancerous) or by defining a change in one or more feature space dimensions (e.g., increasing

elongation), and the disclosed technology may be used to generate images illustrating changes in the cell's appearance as it moved along that vector. As another example, in some cases a person or entity who was responsible for training the machine learning models which generate representations in feature space based on cell images, may use the disclosed technology to visualize cells generated with embedding values at the limits of what was present in the training data. In this way, by determining if the images created for cells with values at the limits of feature space included artifacts, the user may identify if the quality of the underlying training data may be improved by replacing the currently available images with cleaner images in the areas where artifacts were observed. As yet another example, the disclosed technology may be used to create an atlas, in which synthetic images may serve as references for cells in different disease states and be used for diagnostic applications.

Image Context Integration

[0389] It should be understood that the techniques described above for representing images as locations in feature space may also be applied to feature spaces with dimensions based on data regarding an image. As an example, in some cases a feature space for cell images may include dimensions corresponding to attributes of the imaged cells, such as protein production or gene expression. In such a case, a machine learning model (e.g., neural network such as described above) may be trained using training data which includes not only images, but also metadata corresponding to the additional feature space dimensions such as protein production or gene expression. This type of training data may be obtained, for example, by capturing images of cultured cells and annotating the captured images with information corresponding to the relevant feature space dimensions, such as the protein production or gene expression of the imaged cells at the time the images were captured. Similar approaches can also be applied to other types of information, such as temporal information (e.g., how long had elapsed since the cell was imaged in a default state, such as a healthy state without any adverse conditions), or treatment information (e.g., a type of substance, such as a drug or drug metabolite, the imaged cell had been exposed to). Further variations, such as use of compound dimensions such as amount of time since a treatment with a particular type of drug had begun, are also possible and may be implemented by those of ordinary skill in the art without undue experimentation in light of this disclosure. Accordingly, the examples of dimensions in feature space described herein may be illustrative only, and not limiting.

[0390] Implementations of the disclosed technology which incorporate information such as time, protein production, treatments, etc. may be used to provide a variety of types of information which may be of use in particular situations. For example, a system implemented based on this disclosure may answer a question such as what a cell would look like after a specified period of

time had passed, or after a specified period of time had passed given a particular treatment. In some cases, this type of question may be answered by using information provided by the user (e.g., amount of time which had elapsed, whether there was a treatment regimen in place and, if so, what the regimen consisted of) as conditioning information for generating a synthetic cell image which responded to the user's question using techniques such as described above. As another example, questions like what a cell would look like after a given period of time had passed under a given set of conditions (e.g., a particular drug treatment) could be answered by identifying a vector in feature space (e.g., as may be derived from training data such as described above) showing changes in the cell over time under the given conditions, using that vector to project a location in feature space from an initial condition (e.g., a cell image provided by the user) to the specified future time, and then converting that location into a synthetic image answering the user's question using techniques such as described above. Similarly, questions like how long it would take for a cell to reach a certain state (e.g., no longer being viable) or to reach certain functionality (e.g., gene expression or protein production) could be answered by identifying a location in feature space corresponding to the specified state or functionality, and then measuring a distance along a time dimension in the feature space from that location to the location of a cell in an initial condition (e.g., a cell that had just been cultured). Other uses for, and questions which could be answered using, this type of additional information are also possible and will be immediately apparent to those skilled in the art in light of this disclosure. Accordingly, the description above of potential uses of implementations which use temporal or other information should be understood as being illustrative only, and should not be treated as limiting.

Hardware Integrations

[0391] While the above descriptions have focused generally on algorithms, machine learning models, and other items suitable for software implementation, it is to be understood that some versions which may be implemented based on this disclosure may also incorporate hardware to take advantage of and/or support the algorithms, machine learning models, etc. described previously. For example, in some cases, the generation of cell images, as well as other tasks which involve potentially parallelizable performance of the same operation on different data (e.g., training machine learning models), may be performed with special purpose hardware which is optimized for image rendering and/or parallel operations. As an example of such hardware, consider FIG. 26. That figure depicts a multiprocessor unit 901 which includes an instruction unit 902 to provide single instructions in parallel to each of a set of processors 903-1 to 903-n. These processors may then simultaneously apply the instruction from the instruction unit 902 to a set of different data items from memory 904. In this way, a single instruction may be loaded and applied to multiple

data items in many fewer cycles (e.g., load n data items, apply instruction to each of the n data items), than might be needed if rendering was performed using a conventional central processing unit (CPU) (e.g., load data item 1, apply instruction to data item 1, load data item 2, apply instruction to data item 2, ... load data item n , apply instruction to data item n). This can be useful in generating images, and in training and applying machine learning models such as described, because those activities involve applying the same operations to many different items (e.g., matrix multiplication with different data). Of course, it is to be understood that other hardware optimized for this type of single instruction multiple data processing (e.g., graphics processing unit(s), such as the Arc A750 from Intel Corporation, or the GeForce RTX 4070 from NVIDIA Corporation) may also be used, and that some versions of the disclosed technology may use conventional central processing units rather than the optimized hardware described in this paragraph, and so the example of FIG. 26 is to be understood as being illustrative only, and is not to be treated as limiting.

[0392] As another example of how particular hardware may be integrated into some versions of the disclosed technology, consider the capture of natural images which may be used in image generation as described above. For instance, in some cases, a version of the disclosed technology may include imaging hardware by which a user may capture one or more cell images, as well as software which may integrate that hardware with cell image generation technology such as described, and which may allow a user to review images captured by the imaging system and then select images to provide to the image generation system as inputs. For example, a user may be allowed to ask for images to be generated indicating what may happen if a cell became cancerous, or if it transitioned into another specified state, etc. An example of such an imaging system is illustrated in FIG. 27, as discussed below.

[0393] FIG. 27 shows an example of a cell analysis system 1000, which may be used to capture images of cells. System 1000 of this example includes a pump 1010 that is to drive a sample cell-containing fluid from a reservoir 1012 into a cartridge 1020. Cartridge 1020 may be provided as a modular component, such that cartridge 1020 may be readily replaced within system 1000 (e.g., to analyze different batches of sample cells, etc.). The remaining components of system 1000 that do not get replaced each time cartridge 1020 is replaced may be collectively referred to as “the instrument.” The instrument of system 1000 may include pump 1010; or pump 1010 may be considered as a separate component such that a different pump 1010 may be used when a different batch of sample cells is being analyzed. Similarly, the instrument of system 1000 may include reservoir 1012; or reservoir 1012 may be considered as a separate component such that a different reservoir 1012 may be used when a different batch of sample cells is being analyzed.

[0394] In some examples, reservoir 1012 comprises a syringe barrel; and pump 1010 comprises

a syringe pump. In other examples, pump 1010 may take any other suitable form, including but not limited to a gravity feed, a peristaltic pump, etc. Reservoir 1012 may also take any other suitable form, including but not limited to a vial, tube, etc. The sample in reservoir 1012 may be prepared by fixation and staining; and may contain viable cells. The fluid in which the sample cells are contained may include an aqueous solution (e.g., water, buffer, saline, etc.), an oil, or any other suitable fluid.

[0395] Cartridge 1020 includes a flow channel 1022 fluidically coupled with pump 1010, such that pump 1010 is to drive the sample cell-containing fluid from reservoir 1012 through flow channel 1022. Cartridge 1020 may comprise a structure through which fluid may flow; and through which cells in the fluid may be imaged. A light source 1030 generates light for such imaging. In particular, an optical assembly 1032 directs light from light source 1030 toward an imaging region of flow channel 1022. In some examples, light source 1030 comprises a source of incoherent white light, such as an arc lamp, etc. In other examples, light source 1030 may take any other suitable form, such as a laser. Optical assembly 1032 may comprise any suitable number and/or arrangement of lenses and/or other elements as will be apparent to those skilled in the art in view of the teachings herein.

[0396] The light from light source 1030, as directed by optical assembly 1032, illuminates cells as the cells pass through the imaging region of flow channel 1022. An objective lens assembly 1040 is positioned on the opposite side of the imaging region of flow channel 1022, magnifies the images of cells passing through the imaging region of flow channel 1022, and directs the magnified images to a camera 1042. Objective lens assembly 1040 and camera 1042 thus cooperate to capture high resolution images of cells that pass through the imaging region of flow channel 1022, as illuminated by light source 1030 and optical assembly 1032. By way of example only, objective lens assembly 1040 may provide magnification ranging from approximately $10\times$ to approximately $200\times$. In other examples, objective lens assembly 1040 may provide any other suitable level of magnification. By way of further example only, camera 1042 may provide an exposure time ranging from approximately $0.001\ \mu\text{s}$ to approximately 1 ms. In other examples, camera 1042 may provide any other suitable exposure time. In examples described below, objective lens assembly 1040 and camera 1042 have an optical axis along the z-dimension. As described, images captured using the camera 1042 and lens assembly 1040 may be provided to a computer or other suitable hardware which can use them in image generation or for other purposes.

Additional Variations

[0397] While the above description included various examples and implementations, those examples and implementations are illustrative only, and that other variations may be made and

used by those of skill in the art in light of this disclosure. For instance, when generating images using a process such as shown in FIG. 24, the definition of first and second states in block 701 may include a user providing first and second sets of pictures, and defining the first and second states based on those sets. In such a case, the initial embedding for the first state may be an embedding whose dimensions have values equal to the averages for those dimensions in the first set of images. Additionally, in such a case the determination in block 706 of whether the second state has been reached may be based on whether the most recently created additional prompt was within a convex hull defined by the embeddings of the second set of pictures in feature space. As another example of a potential variation, in some cases, rather than defining the set of prompts through iteration of blocks 705-707, and then subsequently generating images in block 708, an image generation process may generate an image for each prompt as soon as that prompt itself has been created, rather than waiting until the set of prompts is complete. Similarly, while the above description focused on the creation of cell images, the disclosed technology may be applied to other types of images as well, such as images of tumors, or of various parts of the human body created through magnetic resonance imaging (MRI images) Other approaches and variations are also possible, and will be immediately apparent to those of skill in the art in light of this disclosure. Accordingly, the variations described in this section, as well as the examples and implementations set forth elsewhere herein, does not imply limitations on the protection provided by this document or any related document.

Non-Limiting Working Examples

[0398] The following examples are intended to be purely illustrative, and not limiting of the present subject matter. The following examples relate to various non-exhaustive ways in which the teachings herein may be combined or applied. The following examples are not intended to restrict the coverage of any claims that may be presented at any time in this application or in subsequent filings of this application. No disclaimer is intended. The following examples are being provided for nothing more than merely illustrative purposes. It is contemplated that the various teachings herein may be arranged and applied in numerous other ways to achieve the benefits described here. It is also contemplated that some variations may omit certain features referred to in the below examples. Therefore, none of the aspects or features referred to below may be deemed critical unless otherwise explicitly indicated as such at a later date by the inventors or by a successor in interest to the inventors. If any claims are presented in this application or in subsequent filings related to this application that include additional features beyond those referred to below, those additional features shall not be presumed to have been added for any reason relating to patentability.

Example 1 – Evaluation of the human foundation model using different cell lines.

[0399] Cancer cells lines A375 and Caov-3, and immune cell line Jurkat were used to evaluate the classification performance of the human foundation model. Polystyrene beads with a size of 6 micrometers (μm) were used as control. The cell lines and polystyrene beads were imaged using the microfluidics platform (e.g., REM-I platform) as described herein and combined *in silico* to evaluate the performance of the human foundation model. The human foundation model processed the images of the cell lines and polystyrene beads, extracted deep learning and morphometric features. These features were standardized and projected into a lower dimensional principal components analysis (PCA) basis. Nearest neighbors were computed in the PCA space, then used to compute 2D Uniform Manifold Approximation and Projections (UMAP). Table 1 (above) is a panel of deep learning derived features generated using the DL model of the human foundation model and used in the present examples. Table 2 (above) is a panel of morphometric features generated using the computer vision model of the human foundation model and used in the present examples. It should be noted that the number and types of features listed in Tables 1 and 2 are provided only as examples, without limiting the scope of the present disclosure. More and/or different features can be included. This example illustrates generated and extracted features using the human foundation model and computer vision model, including features that can be referred to as blobs or granules.

Example 2 – System for Cell Morphology Analysis.

[0400] Table 3 in this example lists parameters and specifications of an example system such as described with reference to FIG. 2A (e.g., REM-1) for cell morphology analysis as well as image analysis generally, cell sorting, and other operations described herein. Table 5 lists the example components of one example system. “*” in Table 4 denotes the specification is dependent on sample characteristics and/or sorting configurations.

Category	Parameter	Specification
Facilities	Instrument dimensions	H: 29.5 in / 75 cm W: 35.5 in / 90 cm D: 29.5 in / 75 cm
	Included ancillary equipment	Computer tower Monitor Keyboard Mouse
	Electrical	3 × 100-240 V surge-protected outlets
	Network connection	1 Gbps ethernet with >150 Mbps upload bandwidth
	Clean dry air connection	0.55-0.72 MPa / 80-105 psi

	Temperature operation ranges	59-86 °F / 15-30 °C
	Operating relative humidity (%)	15-70, non-condensing
Instrument	Input cell size (pm)	6 - 25
	Output cell viability (% of input viability)	>95*
	Output collection	6x positive outlet wells 1x negative outlet tube 1x waste bottle
Instrument	Positive outlet capacity (cells/well)	Up to 3,000
	Recommended run time (min)	Live cells: Up to 180
		Fixed cells: Up to 600
	Image resolution (pm/pixel)	0.16
	Image size (pixels)	256 × 256
	Imaging throughput (events/s)	Up to 1,000
	Sorting throughput (cells/s)	Up to 30

TABLE 5. Components of the REM-1 system.

Name	Description
REM-I Instrument	Microfluidic instrument
REM-I Imaging Kit	Reagents and consumables for imaging workflow
REM-I Sorting Kit	Reagents and consumables for imaging plus sorting workflow
Human Foundation Model (HFM)	Artificial intelligence (AI) model for high-dimensional single-cell morphology analysis
Data Suite	Data suite for visualizing, analyzing, and storing data

Example 3

[0401] A method comprising: generating a set of prompts, wherein each prompt of the set of prompts comprises a set of values corresponding to characteristics of an image type; and generating a set of images, wherein generating the set of images comprises, for each prompt from the set of prompts, generating an image corresponding to that prompt from at least providing that prompt to a trained generative artificial intelligence model; and displaying a user interface to present the set of images.

Example 4

[0402] The method of example 3, wherein: generating the set of prompts comprises generating the set of prompts in a sequence, wherein each prompt after an initial prompt in the sequence is generated by performing one or more modification acts, wherein the one or more modification acts comprise modifying one or more values of the set of values comprised by a preceding prompt in

the sequence; and the user interface is to present the set of images in the sequence.

Example 5

[0403] The method of example 4, wherein: the method comprises identifying a set of visual characteristics which differ between the initial prompt in the sequence and one or more other prompts in the sequence; and the method comprises, for each of one or more images from the set of images, displaying, using the user interface, highlighted features corresponding to the set of visual characteristics as they appear in that image.

Example 6

[0404] The method of example 5, wherein: for each prompt from the set of prompts, the set of values comprised by that prompt comprises at least one value corresponding to a morphometric feature; and the set of visual characteristics comprises at least one visual characteristic that is not a morphometric feature having a value included in any prompt from the set of prompts.

Example 7

[0405] The method of any of examples 4-6, wherein: the method further comprises receiving user input data; the sequence of prompts establishes a transition path from a first state to a second state, wherein both the first state and the second state are determined using at least the user input data; the method further comprises identifying a subset of the set of values as significant values separating the first state from the second state; and for each prompt from the set of prompts after the initial prompt, the one or more modification acts are performed by modifying one or more of the subset of the set of values identified as significant values separating the first state from the second state.

Example 8

[0406] The method of any of examples 4-6, wherein: the method comprises receiving user input data; and for each prompt after the initial prompt, the one or more modification acts are performed by identifying a location along a vector in a space defined by the characteristics of the image type, wherein: the location is more distant from a location for the initial prompt in the sequence than a location for the preceding prompt in the sequence; and the vector is determined using at least the user input data.

Example 9

[0407] The method of example 8, wherein: the user input data identifies a subset of the set of values as modifiable values; and every location along the vector in the space defined by the set of values differs from every other location along the vector in the space defined by the set of values in changes to the modifiable values.

Example 10

[0408] The method of any examples 8-9, wherein: the set of values comprises a value for a bounded characteristic, wherein the bounded characteristic is a characteristic of the image type for which there is a logical upper bound on values for the bounded characteristic; and the set of prompts comprises at least one prompt in which the value corresponding to the bounded characteristic is greater than the logical upper bound on values for the bounded characteristic.

Example 11

[0409] The method of any of examples 8-10, wherein the method comprises, for one or more images from the set of images, displaying, via the user interface, highlighting on portions of that image which differ from another image from the set of images.

Example 12

[0410] The method of example 8, wherein: the user input data comprises an identification of a time period, and a context for change over time; and the vector is a direction in the space defined by the characteristics of the image type for changes over time given the context from the user input data.

Example 13

[0411] The method of any of examples 8-12, wherein the space defined by the characteristics of the image type has dimensions comprising a temporal dimension.

Example 14

[0412] The method of any of examples 3-13, wherein generating the set of images comprises, for each prompt from the set of prompts: obtaining a derived prompt by providing that prompt as input to a latent diffusion model; and providing the derived prompt as input to a decoder.

Example 15

[0413] The method of any of examples 3-13, wherein generating the set of images comprises, for each prompt from the set of prompts, providing that prompt to a decoder.

Example 16

[0414] The method of any of examples 3-13, wherein generating the set of images comprises, for each prompt from the set of prompts, providing that prompt to a conditional generative adversarial network.

Example 17

[0415] The method of any of examples 3-13, wherein, for each prompt from the set of prompts, the values comprised by that prompt comprise values corresponding to: morphometric features; feature space dimensions determined by a machine learning model; or a combination thereof.

Example 18

[0416] The method of any of examples 3-17, wherein generating the set of images further comprises using special purpose hardware to simultaneously apply a single instruction to multiple items of data.

Example 19

[0417] The method of any of examples 3-18, wherein: the set of images comprises a set of subsets of images, wherein each subset of images corresponds to a condition; and generating the set of prompts comprises, for each subset of images, generating a set of prompts whose values are consistent with the condition corresponding to that subset of images.

Example 20

[0418] The method of any of examples 3-19, wherein the image type is selected from a group consisting of tumor images and magnetic resonance imaging (MRI) images.

Example 21

[0419] The method of any of examples 3-19, wherein the image type is cell images, and each image from the set of images is a synthetic cell image.

Example 22

[0420] The method of example 3-21, wherein the method further comprises capturing, using a cell analysis system, a plurality of images of cells from a sample; the method comprises receiving user input data which comprises a selected image from the plurality of images of cells; and the method further comprises generating at least one prompt from the set of prompts by providing the selected image from the plurality of cell images to an encoder.

Example 23

[0421] The method of any of examples 21-22 when taken in combination with any of examples 6-11, wherein the space defined by the characteristics of the image type has dimensions comprising a protein production dimension, and a gene expression dimension.

Example 24

[0422] The method of any of examples 3-23, wherein the set of images comprises a plurality of images, and the set of prompts comprises a plurality of prompts.

Example 25

[0423] A method comprising: determining an initial prompt by, using an encoder, determining an embedding comprising, for each of one or more dimensions of a feature space for images of an image type, a real number value for that dimension which is consistent with a first state; and determining a set of transitional prompts, wherein each prompt from the set of transitional prompts has a preceding neighbor, which is the initial prompt for one of the prompts in the set of transitional

prompts, and which is otherwise another prompt in the set of transitional prompts; and generating a set of images using at least, for the initial prompt, and for each transitional prompt from the set of transitional prompts, generation of an image using at least that prompt and a decoder trained to generate images of the image type based on embeddings in the feature space for images of the image type; and presenting the set of images to a user using a user interface.

Example 26

[0424] The method of example 25, wherein: the method further comprises identifying a set of visual characteristics which differ between the initial prompt and one or more other prompts from the set of transitional prompts; and the method further comprises, for one or more images from the set of images, displaying, using the user interface, highlighted features corresponding to the set of visual characteristics as the highlighted features appear in that image.

Example 27

[0425] The method of example 26, wherein: the set of dimensions in the feature space for images of the image type comprises one or more morphometric features; and the set of visual characteristics comprises at least one visual characteristic which is not a dimension from the set of dimensions in the feature space for images of the image type.

Example 28

[0426] The method of any of examples 25-27, wherein: the method further comprises receiving user input data; the set of transitional prompts establishes a transition path from the first state to a second state, wherein both the first state and the second state are based on the user input data; the method further comprises identifying a subset of the set of dimensions in the feature space for images of the image type as significant dimensions having values separating the first state from the second state; and for each prompt from the set of transitional prompts, determining that prompt comprises by modifying one or more of the subset of the set of dimensions identified as significant dimensions relative to that transitional prompt's preceding neighbor.

Example 29

[0427] The method of any of examples 25-27, wherein: the method further comprises receiving user input data; and determining the set of transitional prompts comprises, for each prompt from the set of transitional prompts, identifying a location along a vector in the feature space, wherein: the location is more distant from a location for the initial prompt in the feature space; and the vector using at least the user input data.

Example 30

[0428] The method of example 29, wherein: the user input data identifies a subset of the set of dimensions in feature space as modifiable dimensions; and every location along the vector in the

feature space differs from every other location along the vector in the feature space only in changes to the modifiable dimensions.

Example 31

[0429] The method of any examples 29-30, wherein: the set of dimensions in the feature space comprises a bounded dimension, wherein the bounded dimension corresponds to a characteristic of the image type for which there is a logical upper bound on values for the characteristic; and the set of prompts comprises at least one prompt in which the value corresponding to the bounded dimension is greater than the logical upper bound on values for the characteristic.

Example 32

[0430] The method of any of examples 29-31, wherein: the user input data comprises an identification of a time period, and a context for change over time; and the vector is a direction in feature space for changes over time given the context from the user input data.

Example 33

[0431] The method of any of examples 25-32, wherein generating the set of images comprises, for the initial prompt and each prompt from the set of transitional prompts: obtaining a derived prompt by providing that prompt as input to a latent diffusion model; and providing the derived prompt as input to a decoder.

Example 34

[0432] The method of any of examples 25-32, wherein generating the set of images comprises, for the initial prompt and each prompt from the set of transitional prompts, providing that prompt to a decoder.

Example 35

[0433] The method of any of examples 25-32, wherein generating the set of images comprises, for the initial prompt and each prompt from the set of transitional prompts, providing that prompt to a conditional generative adversarial network.

Example 36

[0434] The method of any of examples 25-32, wherein the dimensions of the feature space for images of the image type comprise: morphometric features; feature space dimensions determined by a machine learning model; or combinations thereof.

Example 37

[0435] The method of any of examples 25-36, wherein generating the set of images further comprises using special purpose hardware to simultaneously apply a single instruction to multiple items of data.

Example 38

[0436] The method of any of claims 25-37, wherein: the set of images comprises a set of subsets of images, wherein each subset of images corresponds to a condition; and determining the first prompt and each prompt from the set of transitional prompts comprises, for each subset of images, generating a set of prompts whose values are consistent with the condition corresponding to that subset of images.

Example 39

[0437] The method of any of examples 25-38, wherein the image type is selected from a group consisting of: tumor image and magnetic resonance imaging (MRI) images.

Example 40

[0438] The method of any of examples 25-38, wherein the image type is cell images, and each image from the set of images is a synthetic cell image.

Example 41

[0439] The method of example 40, wherein: the method further comprises capturing, using a cell analysis system, a plurality of images of cells from a sample; the method further comprises receiving user input data which comprises a selected image from the plurality of images of cells; and the method further comprises determining the initial prompt by providing the selected image from the plurality of cell images to an encoder.

Example 42

[0440] The method of any of examples 40-41, wherein the one or more dimensions of the feature space for images of the image type comprise a protein production dimension, and a gene expression dimension.

Example 43

[0441] The method of any of examples 29-42, wherein the one or more dimensions of the feature space for images of the image type comprises a temporal dimension.

Example 44

[0442] A non-transitory computer readable medium having stored thereon instructions to perform the method of any of examples 3-43

Example 45

[0443] A system comprising: a display; and the non-transitory computer readable medium of example 44.

Miscellaneous

[0444] While the examples provided above include cells or beads as particles, the teachings herein may be readily applied to other contexts where other kinds of particles are used in addition

to or in lieu of cells or beads.

[0445] The foregoing description is provided to enable a person skilled in the art to practice the various configurations described herein. While the subject technology has been particularly described with reference to the various figures and configurations, these are for illustration purposes only, not limitation. The subject matter described herein is not limited in its application to the details of construction and the arrangement of components set forth in the description herein or illustrated in the drawings hereof. The subject matter described herein is capable of other implementations and of being practiced or of being carried out in various ways.

[0446] While certain examples of the present subject matter have been shown and described herein, it will be obvious to those skilled in the art that such examples are purely illustrative. It is not intended that the invention be limited by the specific examples provided within the specification. While nonlimiting examples of the invention has been described with reference to the aforementioned specification, the descriptions and illustrations of the examples herein are not meant to be construed in a limiting sense. Numerous variations, changes, and substitutions will now occur to those skilled in the art without departing from the invention. Furthermore, it shall be understood that all aspects of the invention are not limited to the specific depictions, configurations or relative proportions set forth herein which depend upon a variety of conditions and variables. It should be understood that various alternatives to the examples described herein can be employed in practicing the invention. It is therefore contemplated that the claims shall also cover any such alternatives, modifications, variations, or equivalents.

CLAIMS

What is claimed is:

1. A method of processing, comprising:
 - extracting, using a Deep Learning (DL) model, a set of machine learning (ML)-based features from a cell image;
 - generating, using the DL model and the set of ML-based features, a plurality of DL embeddings orthogonal to each other;
 - extracting, using a computer vision model, a set of cell morphometric features from the cell image, the cell morphometric features being orthogonal to the plurality of DL embeddings; and
 - generating a plurality of morphometric predictive embeddings using the set of ML-based features and the set of cell morphometric features.
2. The method of Claim 1, wherein the generating the plurality of morphometric predictive embeddings comprises:
 - generating, using the computer vision model and the set of the cell morphometric features, a plurality of morphometric embeddings; and
 - concatenating, using the DL model, the plurality of morphometric embeddings and the plurality of DL embeddings to a plurality of predictive multidimensional vectors comprising predictive probabilities morphometric features.
3. The method of Claim 1, wherein the DL model is trained using a loss function comprising invariance, variance, covariance, morphometric decorrelation terms, or combinations thereof.
4. The method of Claim 1, further comprising:
 - predicting, using the DL model and the plurality of morphometric predictive embeddings, a plurality of morphometric blob features using a correlation coefficient greater than approximately 0.9.
5. The method of Claim 1, further comprising:
 - predicting, using the DL model and the plurality of morphometric predictive embeddings, a plurality of morphometric blob features using a correlation coefficient ranging between approximately 0.85 and approximately 0.95.

6. The method of Claim 1, wherein the set of cell morphometric features comprises a plurality of blob features.

7. The method of Claim 1, wherein the generating the plurality of morphometric predictive embeddings is in a high throughput setting.

8. The method of Claim 1, wherein the plurality of DL embeddings comprises cell morphology information independent from a fixed set of rule-based morphometric features.

9. The method of Claim 1, wherein the plurality of DL embeddings comprises cell morphology data orthogonal to a fixed set of rule-based morphometric features.

10. The method of Claim 1, further comprising:
separating, using the DL model, a plurality of de-correlated DL embeddings and the plurality of morphometric predictive embeddings into different dimensions.

11. The method of Claim 1, wherein the DL model is a self-supervised machine learning (SSL) system, the method further comprising:

de-correlating, using the DL model, the set of morphometric features from the DL embeddings such that the DL model is trained to acquire information not covered using the computer vision model.

12. The method of Claim 1, wherein the cell image comprises a label free image.

13. The method of Claim 1, further comprising:
hosting the DL model and the computer vision model in a cloud computing environment.

14. The method of Claim 1, wherein the method is performed in a cloud computing environment.

15. The method of Claim 1, further comprising:
generating an instruction to sort a cell of the cell image based on the plurality of morphometric predictive embeddings.

16. The method of Claim 1, further comprising:
sorting, using the plurality of morphometric predictive embeddings, a cell of the cell image;
and

feeding data from the sorting back to the DL model in order to train the DL model for future generating of the plurality of DL embeddings.

17. A method for assessing image data, comprising:
extracting, using a Deep Learning (DL) model and image data of a plurality of cells, a vector for a cell of the plurality of cells, the vector comprising a set of machine-learning (ML)-based features and a set of cell morphometric features extracted using a computer vision model;
and

generating, using the DL model and using the set of ML-based features, a plurality of DL embeddings orthogonal to each other.

18. The method of Claim 17, wherein the DL model is trained using a loss function comprising invariance, variance, covariance, morphometric decorrelation terms, or combinations thereof.

19. The method of Claim 17, further comprising:
generating, using the DL model, a plurality of morphometric predictive embeddings using the set of ML-based features and the set of cell morphometric features.

20. The method of Claim 19, wherein the generating the plurality of morphometric predictive embeddings comprises:

generating, using the computer vision model and at least the set of the cell morphometric features, a plurality of morphometric embeddings; and

concatenating, using the DL model, the plurality of morphometric embeddings and the plurality of DL embeddings to a plurality of predictive multidimensional vectors comprising predictive probabilities morphometric features.

21. The method of Claim 19, further comprising:
predicting, using the DL model and at least the plurality of morphometric predictive embeddings, a plurality of morphometric blob features using a correlation ranging between approximately 0.85 to approximately 0.95.

22. The method of Claim 17, wherein the plurality of DL embeddings comprise cell morphology information orthogonal to the set of morphometric features, and wherein the set of morphometric features are determined using a fixed set of rules.

23. The method of Claim 17, further comprising:
separating, using the DL model, a plurality of de-correlated DL embeddings and the plurality of morphometric predictive embeddings into different dimensions.

24. The method of Claim 17, wherein the DL model is a self-supervised machine learning (SSL) system, the method comprising:
de-correlating, using the DL model, the set of morphometric features from the plurality of DL embeddings so that the DL model is trained to acquire information not covered using the computer vision model.

25. The method of Claim 17, wherein the image data comprises a label free image of each cell of the plurality of cells.

26. The method of Claim 17, further comprising: hosting the DL model and the computer vision model in a cloud computing environment.

27. The method of Claim 17, further comprising:
generating an instruction to sort the plurality of cells using the plurality of DL embeddings.

28. The method of Claim 17, further comprising:
sorting, using the plurality of morphometric predictive embeddings, a cell of the cell image;
and
feeding data from the sorting to the DL model in order to train the DL model for future generating of the plurality of DL embeddings.

29. A system for analyzing image data, the system comprising:
at least one processor to execute instructions to perform operations comprising:
extracting, using a Deep Learning (DL) model, a set of machine learning (ML)-based features from a cell image;

generating, using the DL model and using the ML-based features, a plurality of DL embeddings orthogonal to each other;

extracting, using a computer vision model, a set of cell morphometric features from the cell image that are orthogonal to the plurality of DL embeddings; and

generating a plurality of morphometric predictive embeddings using the set of ML-based features and the set of cell morphometric features.

30. The system of Claim 29, wherein the set of cell morphometric features comprises a plurality of blob features.

31. The system of Claim 29, wherein the generating the plurality of morphometric predictive embeddings is in a high throughput setting.

32. The system of Claim 29, wherein the plurality of DL embeddings comprise cell morphology data orthogonal to a fixed set of rule based morphometric features.

33. The system of Claim 29, the operations further comprising:
separating, using the DL model, a plurality of de-correlated DL embeddings and the plurality of morphometric predictive embeddings into different dimensions.

34. The system of Claim 29, wherein the DL model is a self-supervised machine learning (SSL) system, the operations further comprising:

de-correlating the set of morphometric features from the plurality of DL embeddings so that the plurality of DL model is trained to acquire information not covered using the computer vision model.

35. The system of Claim 29, wherein the at least one processor is in a cloud computing environment.

36. A non-transitory computer-readable medium storing instructions that, when executed by processor, cause the processor to perform operations for analyzing image data of a cell image, the operations comprising:

extracting, using a trained Deep Learning (DL) model and from image data a plurality of cells, a vector for a cell of the plurality of cells, the vector comprising a set of machine-learning

(ML)-based features and a set of cell morphometric features extracted by a trained computer vision model;

generating, using the trained DL model and the ML-based features, a plurality of DL embeddings orthogonal to each other; and

extracting, using the trained computer vision model, a set of cell morphometric features from the cell image that are orthogonal to the plurality of DL embeddings.

37. A cloud-based computing system, the system comprising:

at least one cloud-based processor to execute the instructions to perform operations comprising:

extracting, using a Deep Learning (DL) model, a set of machine learning (ML)-based features from a cell image;

generating, using the DL model and using the set of ML-based features, a plurality of DL embeddings orthogonal to each other;

extracting, using a computer vision model, a set of cell morphometric features from the cell image, the cell morphometric features being orthogonal to the plurality of DL embeddings; and

generating a plurality of morphometric predictive embeddings using the set of ML-based features and the set of cell morphometric features.

38. The system of Claim 37, wherein the generating the plurality of morphometric predictive embeddings comprises:

generating, using the computer vision model and at least the set of the cell morphometric features, a plurality of morphometric embeddings; and

concatenating, using the DL model, the plurality of morphometric embeddings and the plurality of DL embeddings to a plurality of predictive multidimensional vectors comprising predictive probabilities morphometric features.

39. The system of Claim 37, wherein the DL model is trained using a loss function comprising one or more invariance, variance, covariance, and morphometric decorrelation terms.

40. The system of Claim 37, the operations further comprising:

predicting, using the DL model and the plurality of morphometric predictive embeddings, a plurality of morphometric blob features using a correlation coefficient greater than approximately 0.9.

41. The system of Claim 37, the operations further comprising:
predicting, using the DL model and the plurality of morphometric predictive embeddings, a plurality of morphometric blob features using a correlation coefficient ranging between approximately 0.85 and approximately 0.95.

42. The system of Claim 37, the operations further comprising:
separating, using the DL model, a plurality of de-correlated DL embeddings and the plurality of morphometric predictive embeddings into different dimensions.

43. The system of Claim 37, wherein the DL model is a self-supervised machine learning (SSL) system, the operations further comprising:

de-correlating, using the DL model, the set of morphometric features from the DL embeddings such that the DL model is trained to acquire information not covered using the computer vision model.

44. A method for cell sorting, comprising:
transporting a cell suspended in a fluid through a flow channel, wherein the flow channel is in fluid communication with a plurality of sub-channels;

capturing one or more images of the cell as the cell is transported through the flow channel;
extracting, by a Deep Learning (DL) model, a set of machine learning (ML)-based features from the one or more images;

generating, using the DL model and the set of ML-based features, a plurality of DL embeddings orthogonal to each other;

extracting, using a computer vision model, a set of cell morphometric features from the one or more images, the cell morphometric features being orthogonal to the plurality of DL embeddings;

generating a plurality of morphometric predictive embeddings using the set of ML-based features and the set of cell morphometric features; and

sorting the cell to a selected sub-channel of the plurality of sub-channels using the plurality of morphometric predictive embeddings.

45. A method for cell sorting, comprising:
transporting a cell suspended in a fluid through a flow channel, wherein the flow channel is in fluid communication with a plurality of sub-channels;

capturing one or more images of the cell as the cell is transported through the flow channel;
extracting, by a Deep Learning (DL) model, a set of machine learning (ML)-based features from the one or more images;

generating, using the DL model and using the set of ML-based features, a plurality of DL embeddings orthogonal to each other; and

sorting the cell to a selected sub-channel of the plurality of sub-channels using the plurality of DL embeddings.

46. A cell sorting system, comprising:

a flow channel configured to transport a cell through the flow channel;

an imaging device configured to capture one or more images of the cell as the cell is transported through the flow channel; and

a processor to execute instructions to perform operations comprising:

extracting, using a Deep Learning (DL) model, a set of machine learning (ML)-based features from a cell image;

generating, using the DL model and using the ML-based features, a plurality of DL embeddings orthogonal to each other;

extracting, using a computer vision model, a set of cell morphometric features from the cell image that are orthogonal to the plurality of DL embeddings;

generating a plurality of morphometric predictive embeddings using the set of ML-based features and the set of cell morphometric features; and

enabling sorting of the cell using one of the plurality of morphometric predictive embeddings, the plurality of DL embeddings, and the set of cell morphometric features.

47. A cell sorting system, comprising:

a flow channel configured to transport a cell through the flow channel;

an imaging device configured to capture one or more images of the cell as the cell is transported through the flow channel; and

a processor to execute instructions to perform operations comprising:

extracting, using a Deep Learning (DL) model and image data of a plurality of cells, a vector for a cell of the plurality of cells, the vector comprising a set of machine-learning (ML)-based features and a set of cell morphometric features extracted using a computer vision model;

generating, using the DL model and using the set of ML-based features, a plurality of DL embeddings orthogonal to each other; and

enabling sorting of the cell using one of the plurality of DL embeddings and the set of cell morphometric features.

48. A method comprising:

generating a set of prompts, wherein each prompt of the set of prompts comprises a set of values corresponding to characteristics of an image type; and

generating a set of images, wherein generating the set of images comprises, for each prompt from the set of prompts, generating an image corresponding to that prompt from at least providing that prompt to a trained generative artificial intelligence model; and

displaying a user interface to present the set of images.

49. The method of claim 48, wherein:

generating the set of prompts comprises generating the set of prompts in a sequence, wherein each prompt after an initial prompt in the sequence is generated by performing one or more modification acts, wherein the one or more modification acts comprise modifying one or more values of the set of values comprised by a preceding prompt in the sequence; and

the user interface is to present the set of images in the sequence.

50. The method of claim 49, wherein:

the method comprises identifying a set of visual characteristics which differ between the initial prompt in the sequence and one or more other prompts in the sequence; and

the method comprises, for each of one or more images from the set of images, displaying, using the user interface, highlighted features corresponding to the set of visual characteristics as they appear in that image.

51. The method of claim 50, wherein:

for each prompt from the set of prompts, the set of values comprised by that prompt comprises at least one value corresponding to a morphometric feature; and

the set of visual characteristics comprises at least one visual characteristic that is not a morphometric feature having a value included in any prompt from the set of prompts.

52. The method of any of claims 48-51, wherein:

the method further comprises receiving user input data;

the sequence of prompts establishes a transition path from a first state to a second state, wherein both the first state and the second state determined using at least the user input data;

the method further comprises identifying a subset of the set of values as significant values separating the first state from the second state; and

for each prompt from the set of prompts after the initial prompt, the one or more modification acts are performed by modifying one or more of the subset of the set of values identified as significant values separating the first state from the second state.

53. The method of any of claims 48-51, wherein:

the method comprises receiving user input data; and

for each prompt after the initial prompt, the one or more modification acts are performed by identifying a location along a vector in a space defined by the characteristics of the image type, wherein:

the location is more distant from a location for the initial prompt in the sequence than a location for the preceding prompt in the sequence; and

the vector is determined using at least the user input data.

54. The method of claim 53, wherein:

the user input data identifies a subset of the set of values as modifiable values; and

every location along the vector in the space defined by the set of values differs from every other location along the vector in the space defined by the set of values in changes to the modifiable values.

55. The method of any claims 53-54, wherein:

the set of values comprises a value for a bounded characteristic, wherein the bounded characteristic is a characteristic of the image type for which there is a logical upper bound on values for the bounded characteristic; and

the set of prompts comprises at least one prompt in which the value corresponding to the bounded characteristic is greater than the logical upper bound on values for the bounded characteristic.

56. The method of any of claims 53-55, wherein the method comprises, for one or more images from the set of images, displaying, via the user interface, highlighting on portions of that image which differ from another image from the set of images.

57. The method of claim 53, wherein:
the user input data comprises an identification of a time period, and a context for change over time; and

the vector is a direction in the space defined by the characteristics of the image type for changes over time given the context from the user input data.

58. The method of any of claims 53-57, wherein the space defined by the characteristics of the image type has dimensions comprising a temporal dimension.

59. The method of any of claims 48-58, wherein generating the set of images comprises, for each prompt from the set of prompts:

obtaining a derived prompt by providing that prompt as input to a latent diffusion model;
and

providing the derived prompt as input to a decoder.

60. The method of any of claims 48-59, wherein generating the set of images comprises, for each prompt from the set of prompts, providing that prompt to a decoder.

61. The method of any of claims 48-60, wherein generating the set of images comprises, for each prompt from the set of prompts, providing that prompt to a conditional generative adversarial network.

62. The method of any of claims 48-61, wherein, for each prompt from the set of prompts, the values comprised by that prompt comprise values corresponding to :

morphometric features;

feature space dimensions determined by a machine learning model, or

a combination thereof.

63. The method of any of claims 48-62, wherein generating the set images further comprises using special purpose hardware to simultaneously apply a single instruction to multiple items of data.

64. The method of any of claims 48-63, wherein:

the set of images comprises a set of subsets of images, wherein each subset of images corresponds to a condition; and

generating the set of prompts comprises, for each subset of images, generating a set of prompts whose values are consistent with the condition corresponding to that subset of images.

65. The method of any of claims 48-64, wherein the image type is selected from a group consisting of: tumor images and magnetic resonance imaging (MRI) images.

66. The method of any of claims 48-65, wherein the image type is cell images, and each image from the set of images is a synthetic cell image.

67. The method of claim 66, wherein:
the method further comprises capturing, using a cell analysis system, a plurality of images of cells from a sample;

the method comprises receiving user input data which comprises a selected image from the plurality of images of cells; and

the method further comprises generating at least one prompt from the set of prompts by providing the selected image from the plurality of cell images to an encoder.

68. The method of any of claims 66-67 when taken in combination with any of claims 6-11, wherein the space defined by the characteristics of the image type has dimensions comprising a protein production dimension, and a gene expression dimension.

69. The method of any of claims 48-68, wherein the set of images comprises a plurality of images, and the set of prompts comprises a plurality of prompts.

70. A method comprising:
determining an initial prompt by, using an encoder, determining an embedding comprising for each of one or more dimensions of a feature space for images of an image type, a real number value for that dimension which is consistent with a first state; and

determining a set of transitional prompts, wherein each prompt from the set of transitional prompts has a preceding neighbor, which is the initial prompt for one of the prompts in the set of transitional prompts, and which is otherwise another prompt in the set of transitional prompts; and

generating a set of images using at least, for the initial prompt, and for each transitional prompt from the set of transitional prompts, generation of an image using at least that prompt and a decoder trained to generate images of the image type based on embeddings in the feature space for images of the image type; and

presenting the set of images to a user using a user interface.

71. The method of claim 70, wherein:

the method further comprises identifying a set of visual characteristics which differ between the initial prompt and one or more other prompts from the set of transitional prompts; and

the method further comprises, for one or more images from the set of images, displaying using the user interface, highlighted features corresponding to the set of visual characteristics as the highlighted features appear in that image.

72. The method of claim 71, wherein:

the set of dimensions in the feature space for images of the image type comprises one or more morphometric features; and

the set of visual characteristics comprises at least one visual characteristic which is not a dimension from the set of dimensions in the feature space for images of the image type.

73. The method of any of claims 70-72, wherein:

the method further comprises receiving user input data;

the set of transitional prompts establishes a transition path from the first state to a second state, wherein both the first state and the second state are based on the user input data;

the method further comprises identifying a subset of the set of dimensions in the feature space for images of the image type as significant dimensions having values separating the first state from the second state; and

for each prompt from the set of transitional prompts, determining that prompt comprises modifying one or more of the subset of the set of dimensions identified as significant dimensions relative to that transitional prompt's preceding neighbor.

74. The method of any of claims 70-73, wherein:

the method further comprises receiving user input data; and

determining the set of transitional prompts comprises, for each prompt from the set of transitional prompts, identifying a location along a vector in the feature space, wherein:

the location is more distant from a location for the initial prompt in the feature space; and

the vector is determined using at least the user input data.

75. The method of claim 74, wherein:

the user input data identifies a subset of the set of dimensions in feature space as modifiable dimensions; and

every location along the vector in the feature space differs from every other location along the vector in the feature space only in changes to the modifiable dimensions.

76. The method of any claims 74-75, wherein:

the set of dimensions in the feature space comprises a bounded dimension, wherein the bounded dimension corresponds to a characteristic of the image type for which there is a logical upper bound on values for the characteristic ; and

the set of prompts comprises at least one prompt in which the value corresponding to the bounded dimension is greater than the logical upper bound on values for the characteristic.

77. The method of any of claims 74-76, wherein:

the user input data comprises an identification of a time period, and a context for change over time; and

the vector is a direction in feature space for changes over time given the context from the user input data.

78. The method of any of claims 70-77, wherein generating the set of images comprises, for the initial prompt and each prompt from the set of transitional prompts:

obtaining a derived prompt by providing that prompt as input to a latent diffusion model; and

providing the derived prompt as input to a decoder.

79. The method of any of claims 70-77, wherein generating the set of images comprises, for the initial prompt and each prompt from the set of transitional prompts, providing that prompt to a decoder.

80. The method of any of claims 70-77, wherein generating the set of images comprises, for the initial prompt and each prompt from the set of transitional prompts, providing that prompt to a conditional generative adversarial network.

81. The method of any of claims 70-77, wherein the dimensions of the feature space for images of the image type comprise:

morphometric features;

feature space dimensions determined by a machine learning model; or

a combination thereof.

82. The method of any of claims 70-81, wherein generating the set of images further comprises using special purpose hardware to simultaneously apply a single instruction to multiple items of data.

83. The method of any of claims 70-82, wherein:

the set of images comprises a set of subsets of images, wherein each subset of images corresponds to a condition; and

determining the first prompt and each prompt from the set of transitional prompts comprises, for each subset of images, generating a set of prompts whose values are consistent with the condition corresponding to that subset of images.

84. The method of any of claims 70-83, wherein the image type is selected from a group consisting of: tumor images and magnetic resonance imaging (MRI) images.

85. The method of any of claims 70-83, wherein the image type is cell images, and each image from the set of images is a synthetic cell image.

86. The method of any of claim 85, wherein:

the method further comprises capturing, using a cell analysis system, a plurality of images of cells from a sample;

the method further comprises receiving user input data which comprises a selected image from the plurality of images of cells; and

the method further comprises determining the initial prompt by providing the selected image from the plurality of cell images to an encoder.

87. The method of any of claims 85-86, wherein the one or more dimensions of the feature space for images of the image type comprise a protein production dimension, and a gene expression dimension.

88. The method of any of claims 70-87, wherein the one or more dimensions of the feature space for images of the image type comprises a temporal dimension.

89. A non-transitory computer readable medium having stored thereon instructions to perform the method of any of claims 48-88.

90. A system comprising:
a display; and
the non-transitory computer readable medium of claim 89.

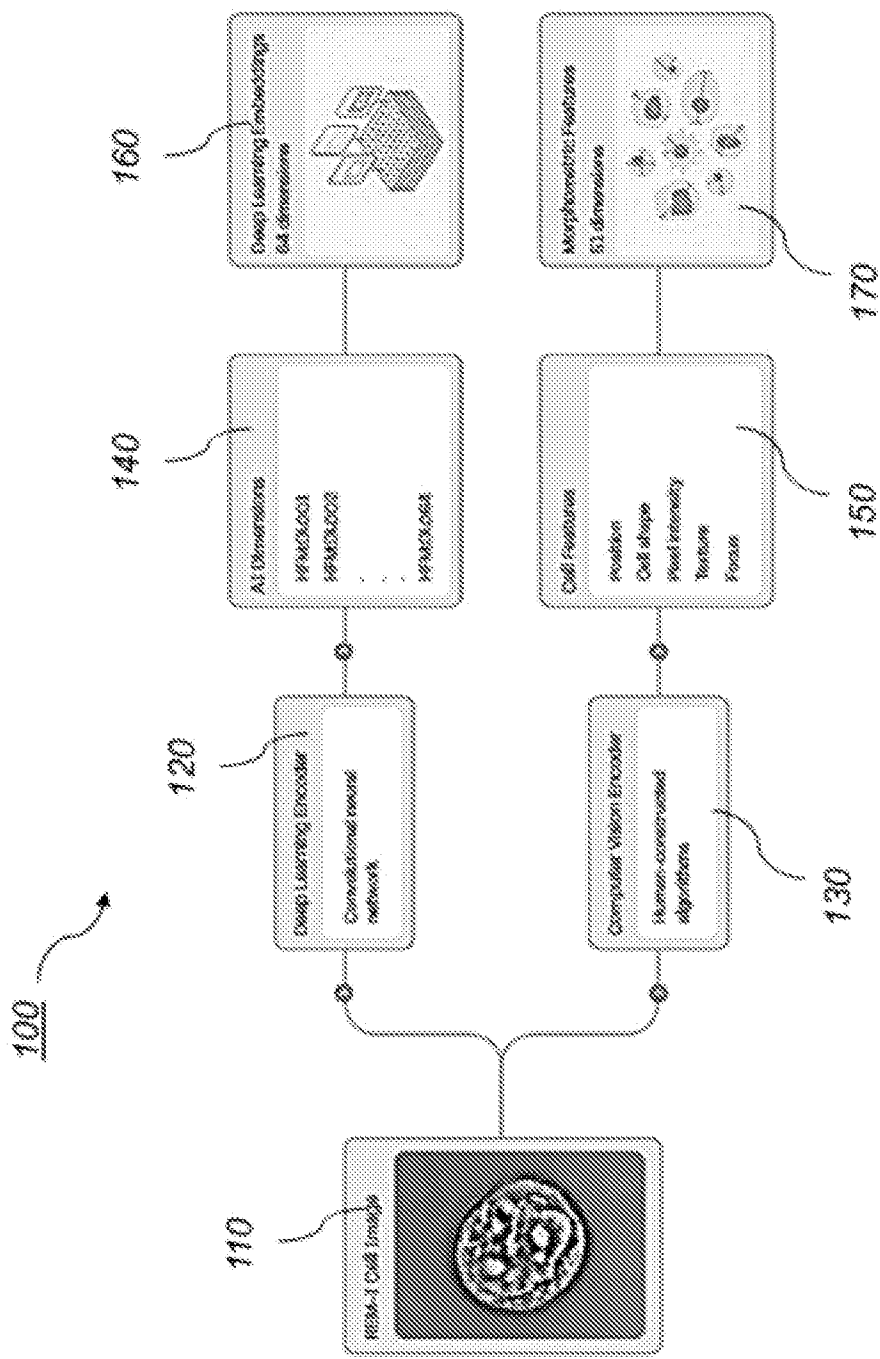


FIG. 1

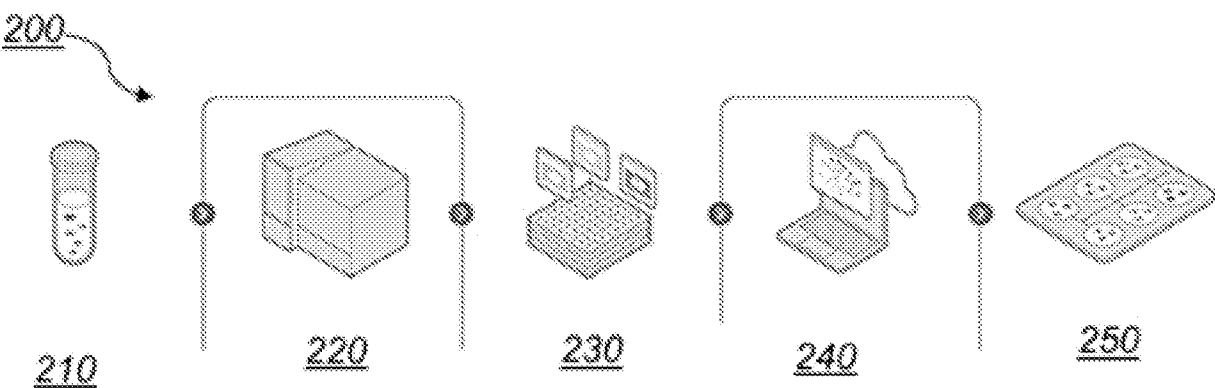
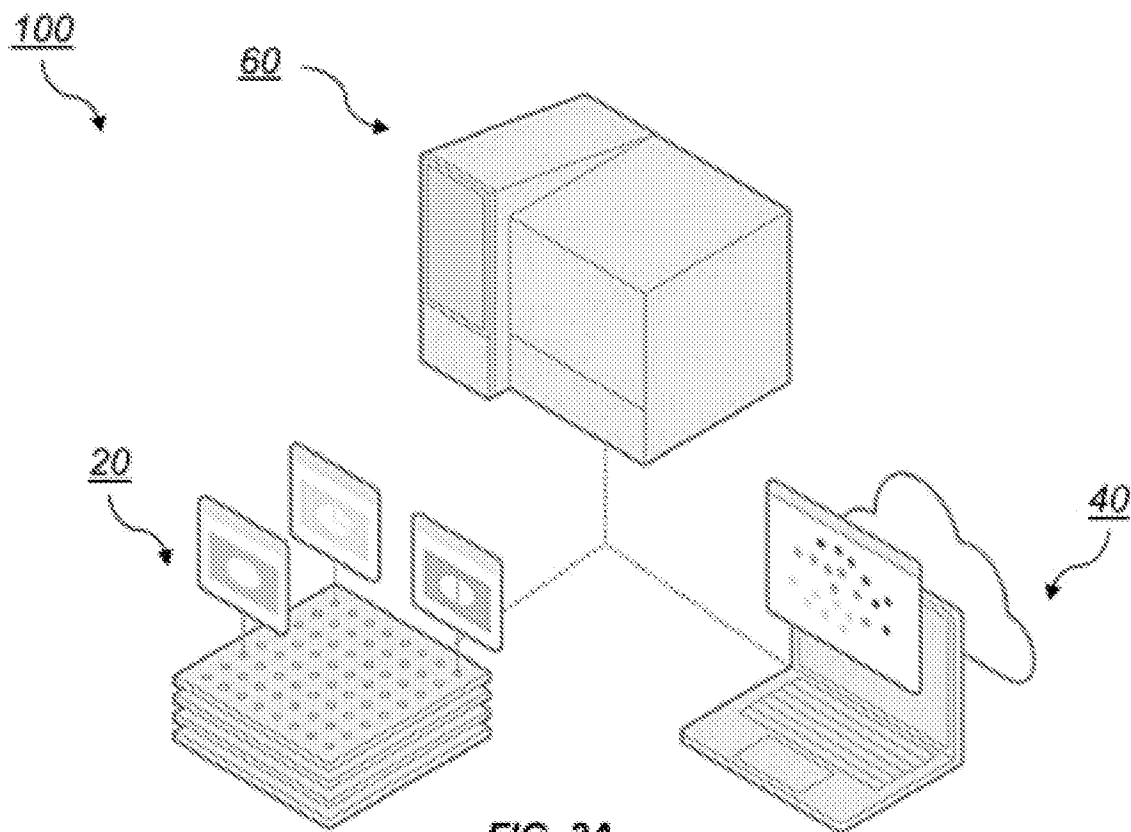


FIG. 2B

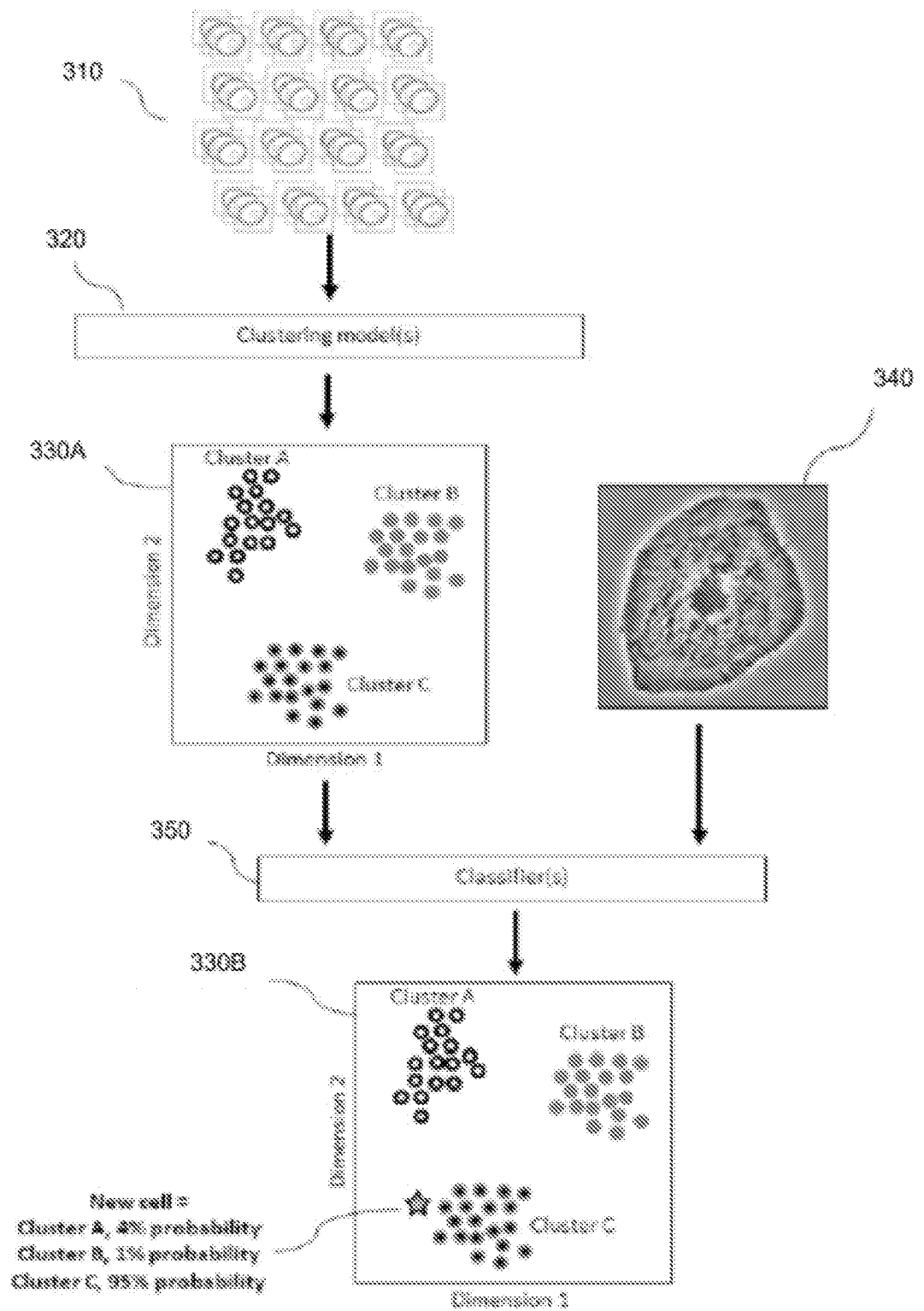


FIG. 3

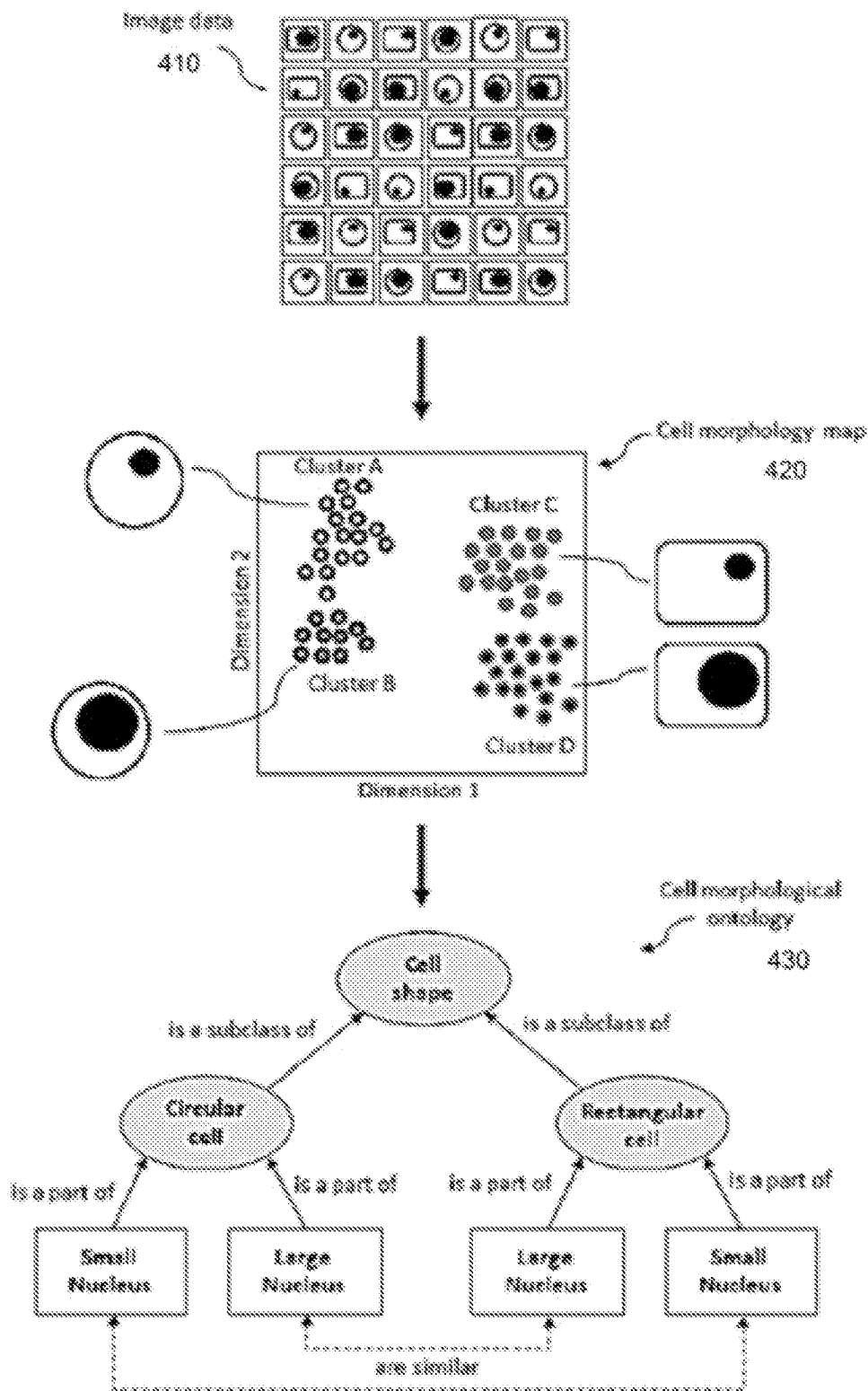


FIG. 4

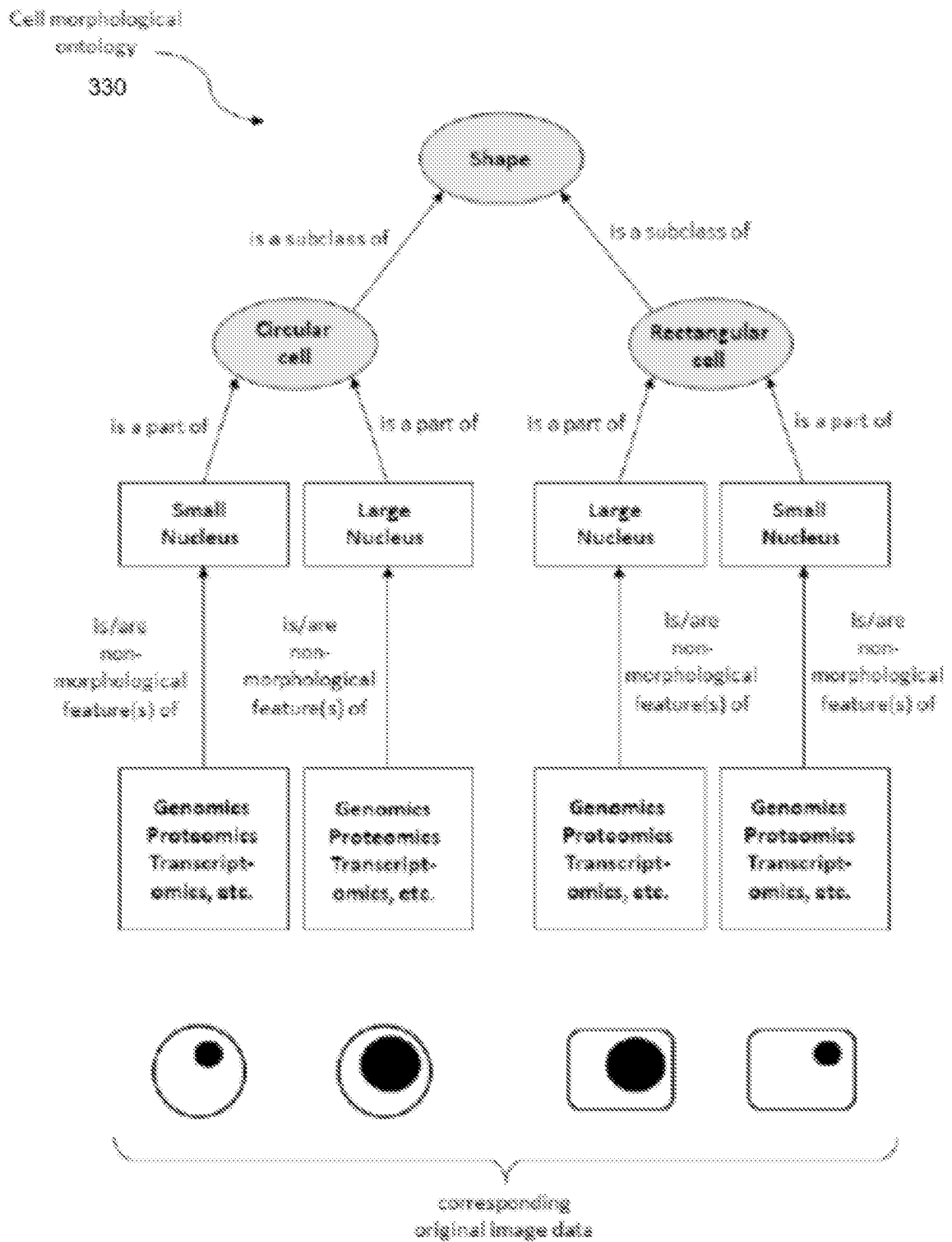


FIG. 5

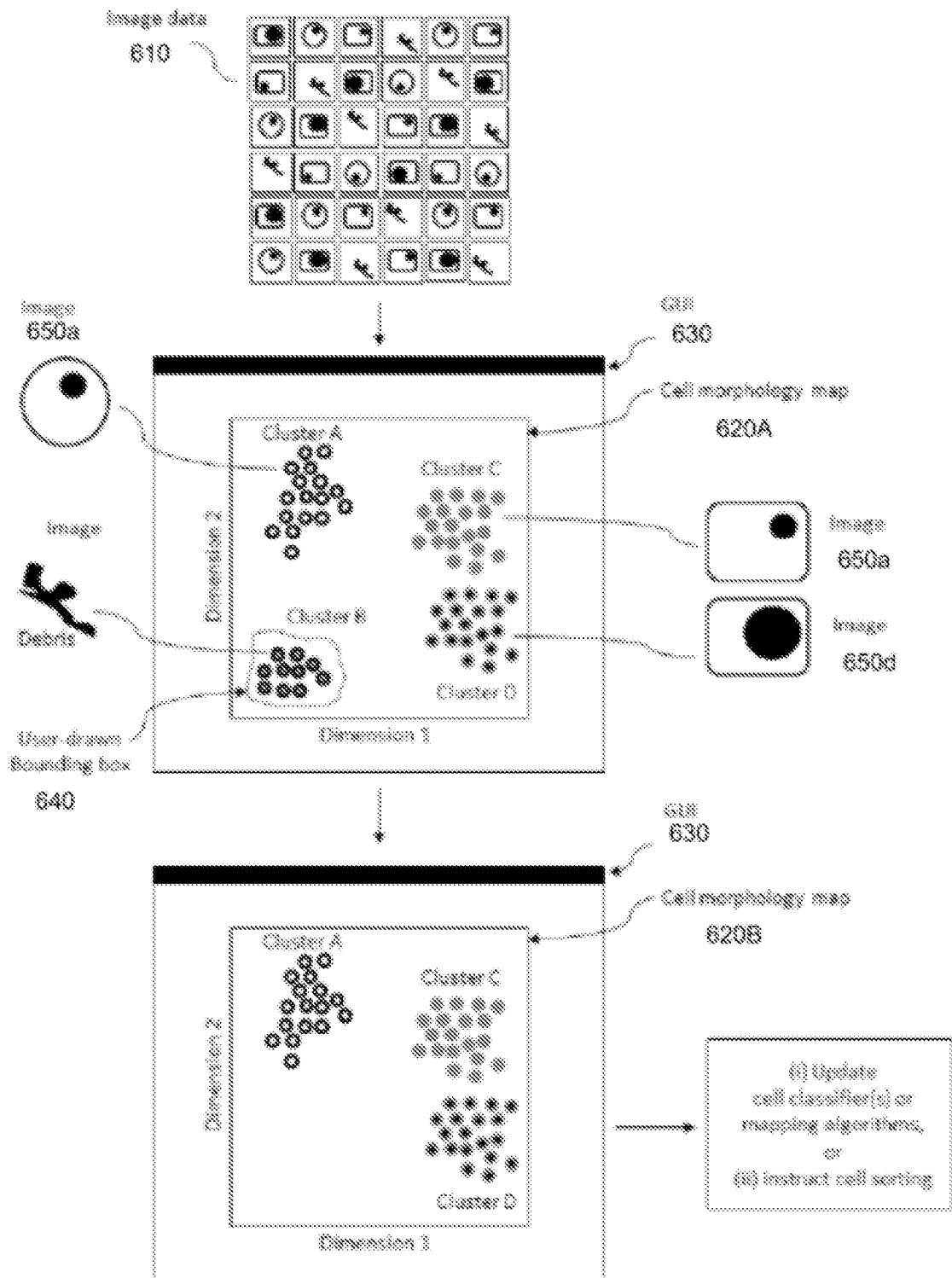


FIG. 6

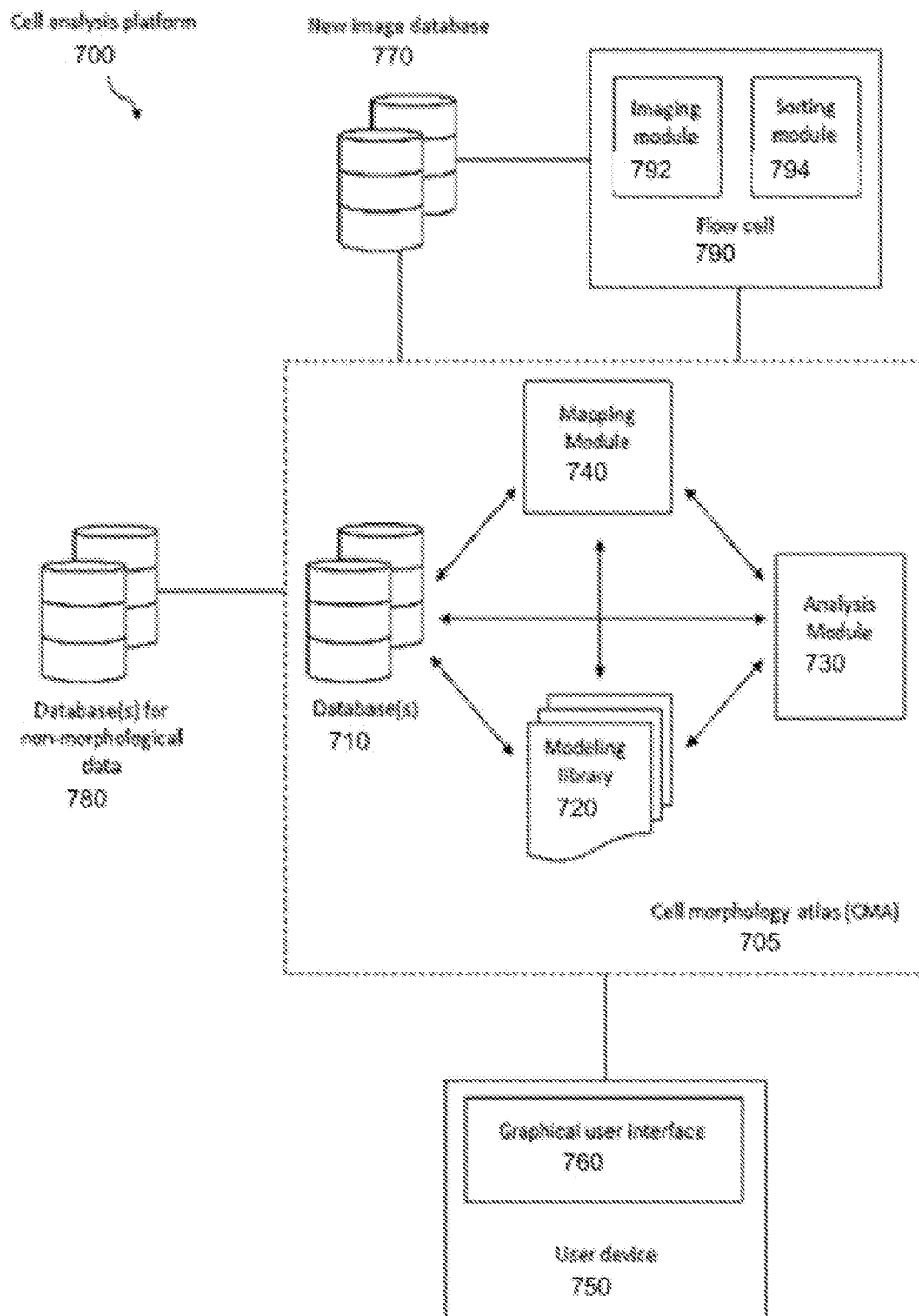


FIG. 7

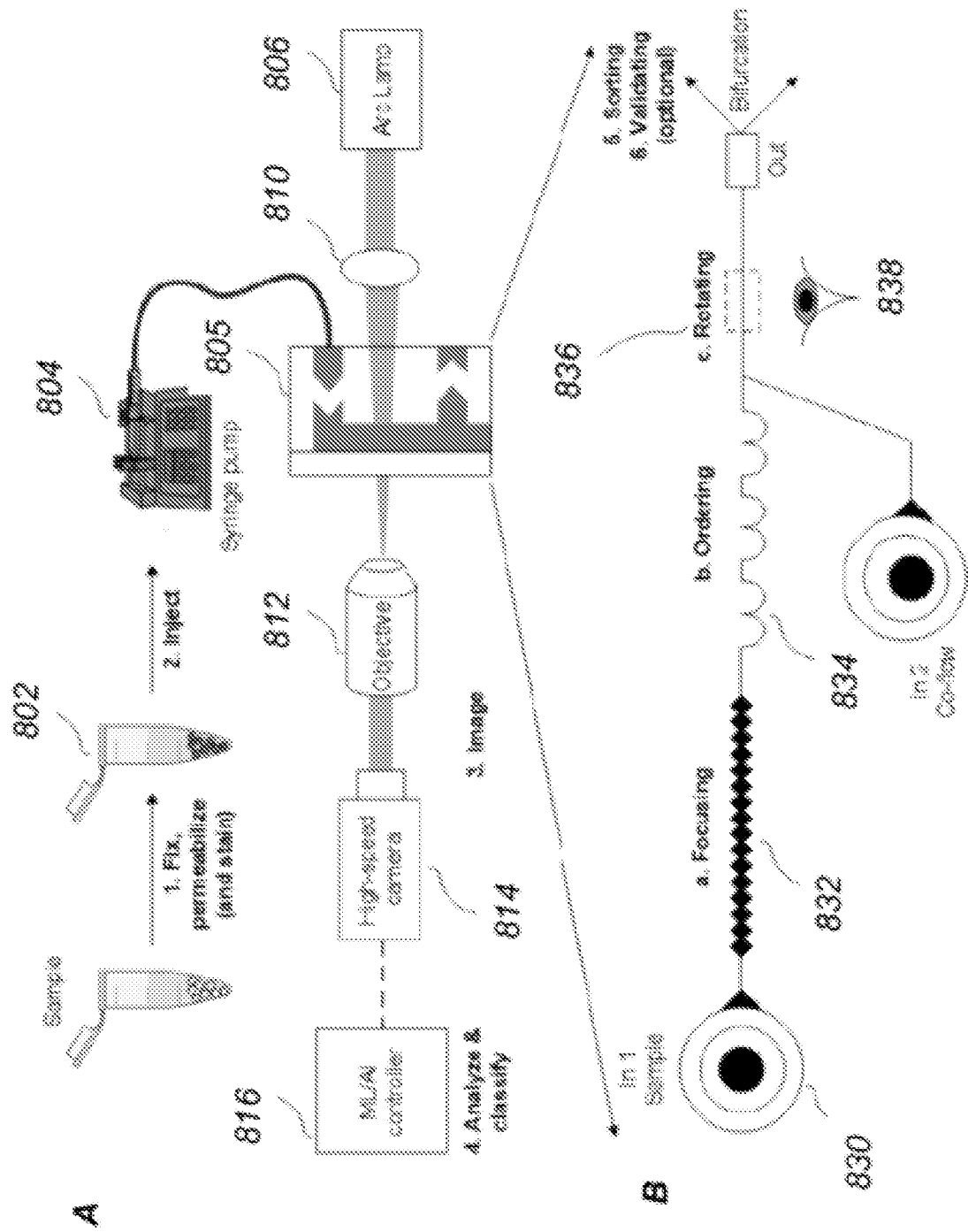
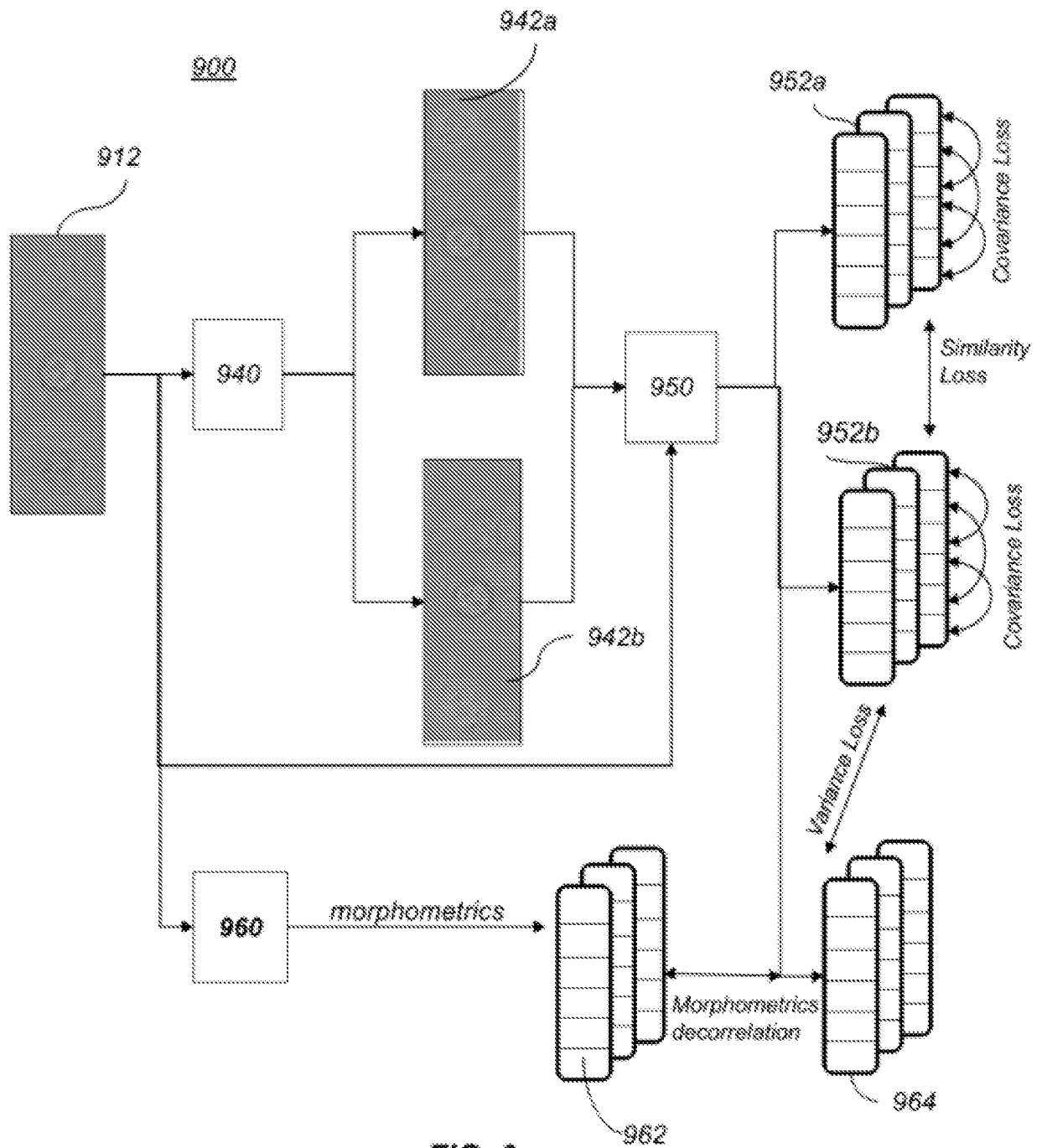


FIG. 8



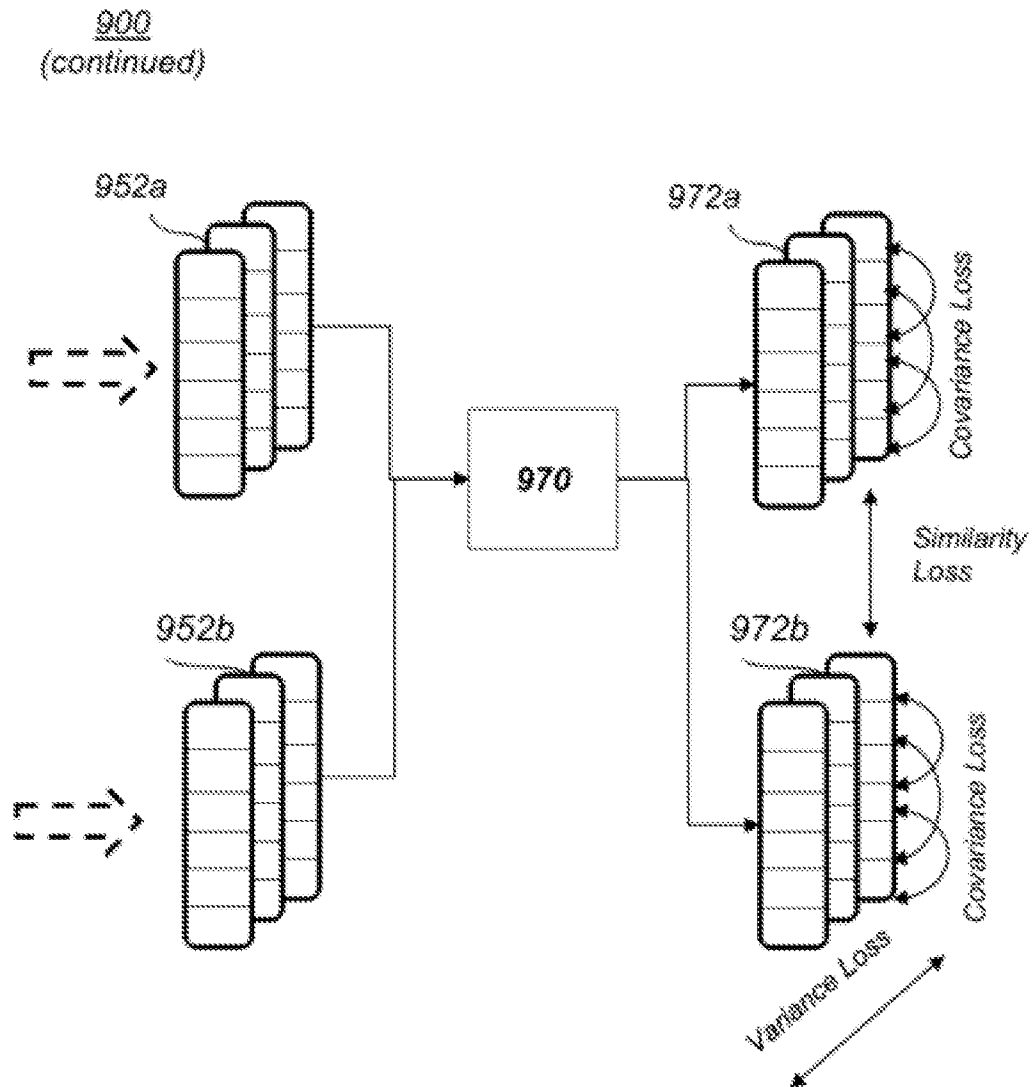
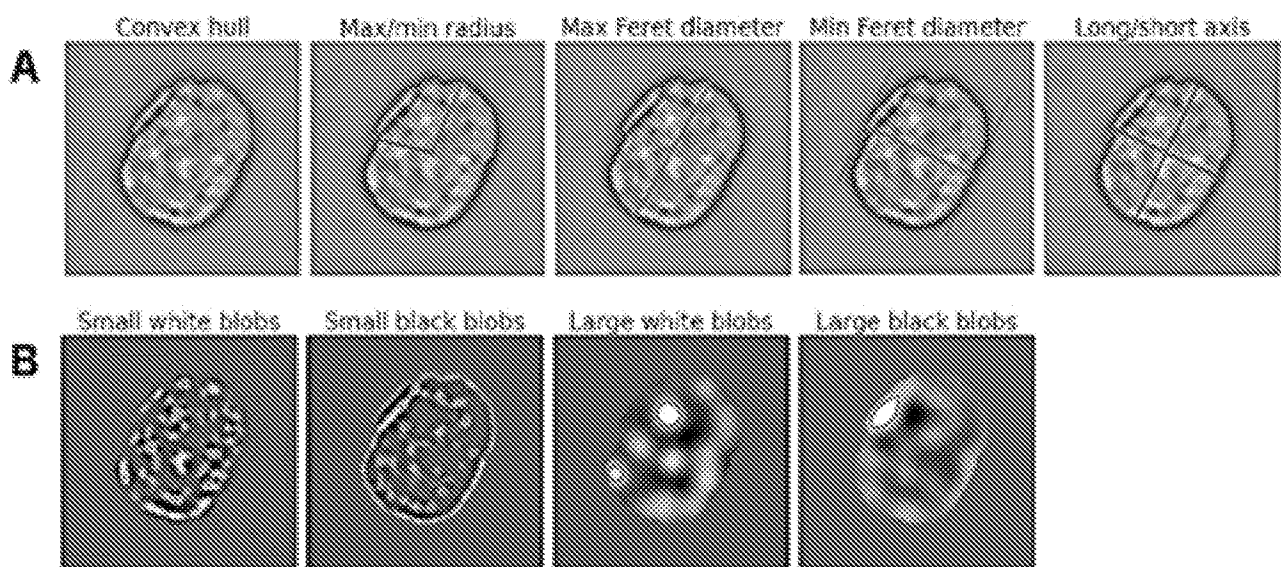


FIG. 10

**FIG. 11**

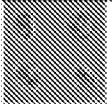
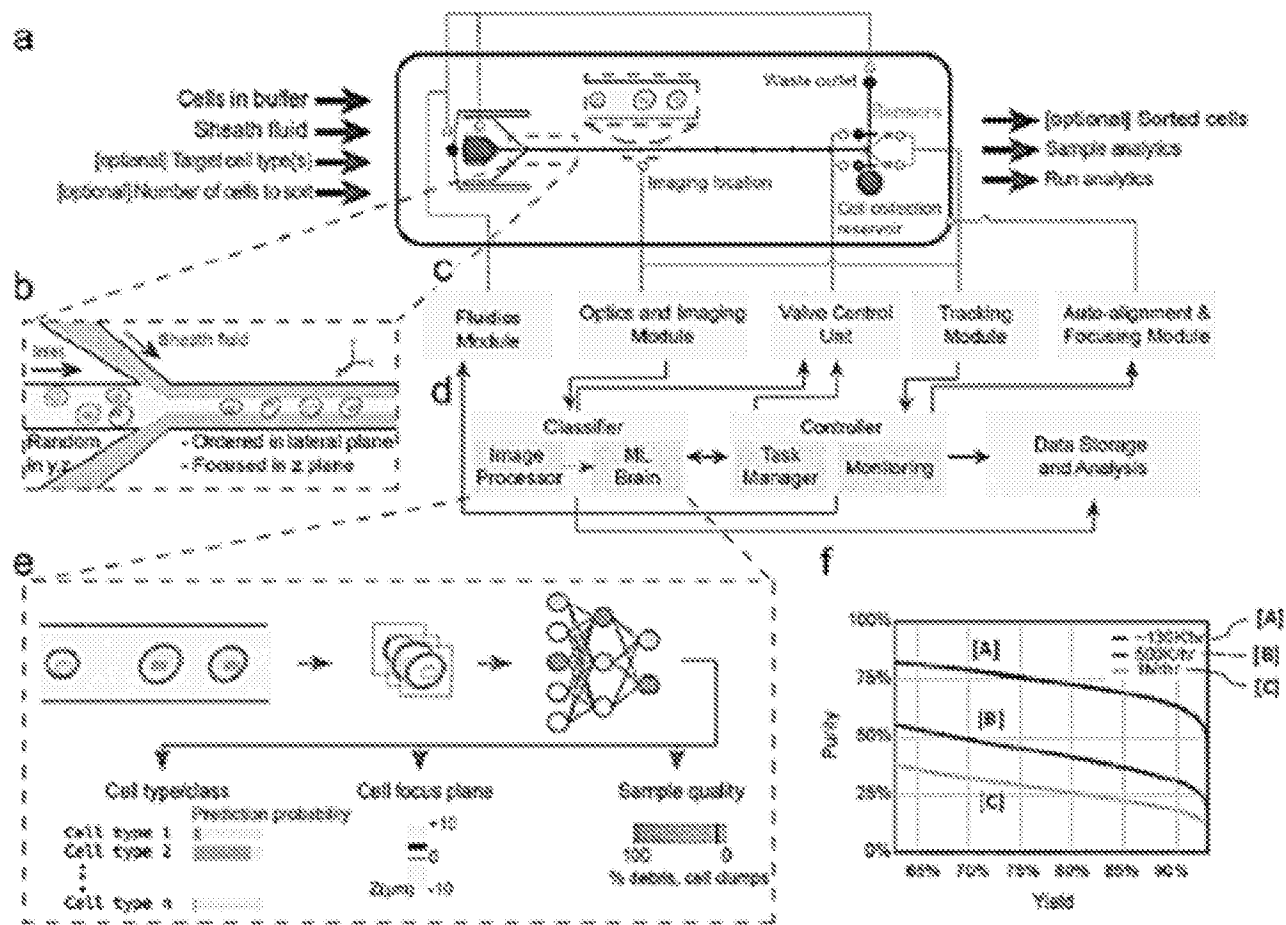
Cell Class	Bead	Jurkat	A375	Caov-3
Training Images	12561	26593	16018	16427
Cell Class	3866	6150	15256	14712
Example Images				

FIG. 12A

True Cell Class	Bead	100%	0.00%	0.00%	0.00%
	Jurkat	0%	90.50%	2.50%	7.00%
	A375	0%	1.10%	89.00%	9.90%
	Caov-3	0%	1.80%	2.70%	85.80%
	Predicted Cell Class				

FIG. 12B



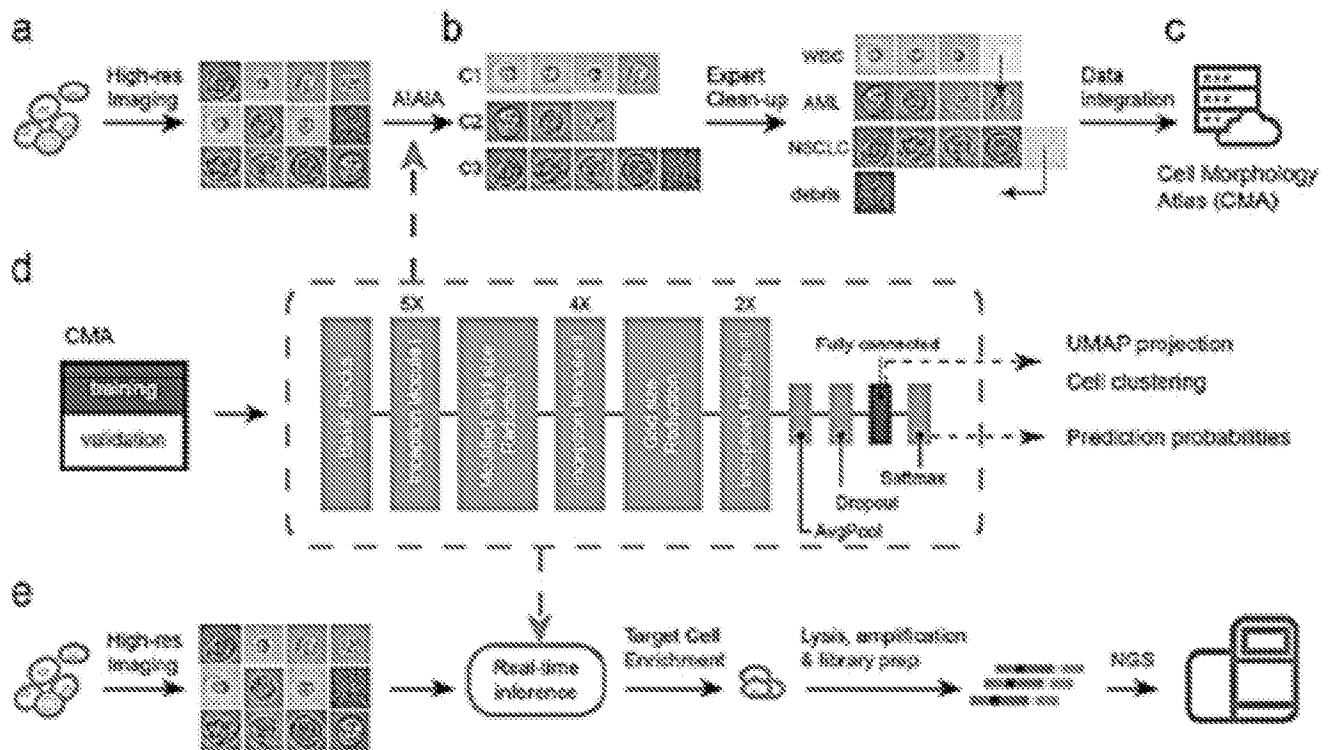


FIG. 14

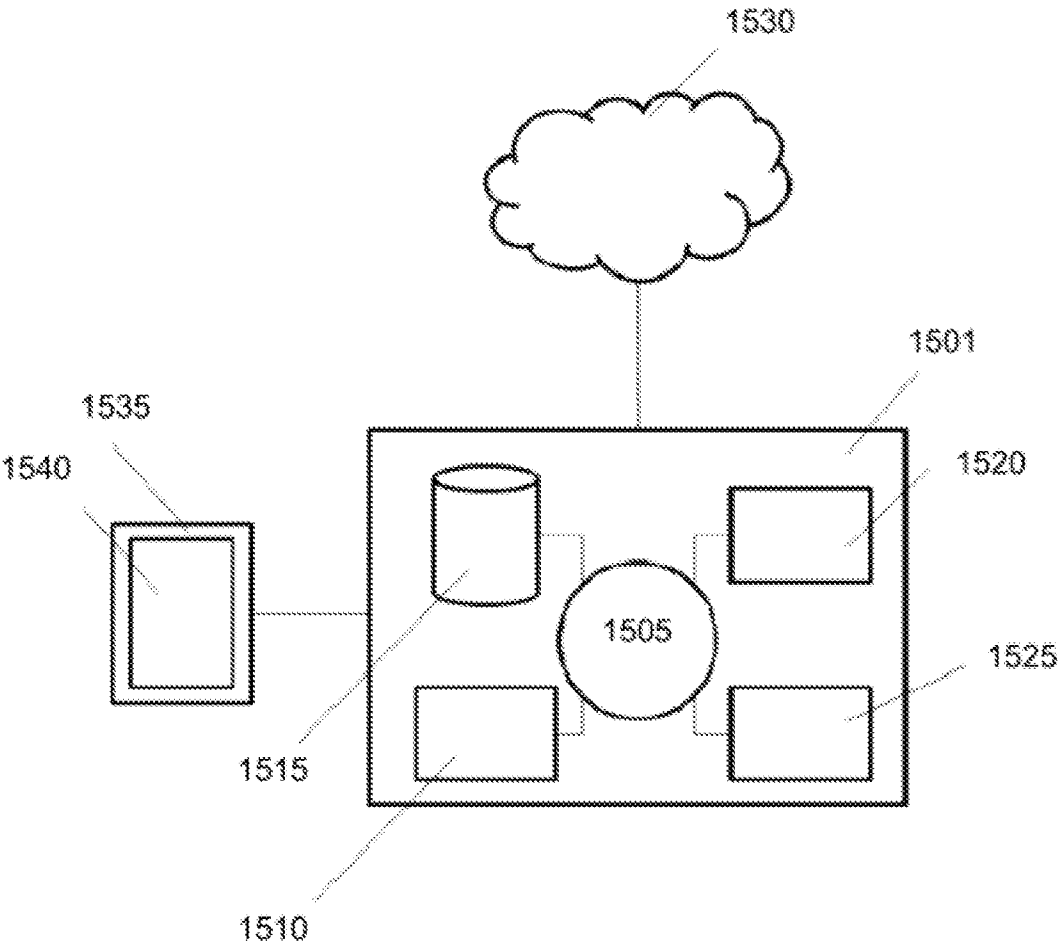
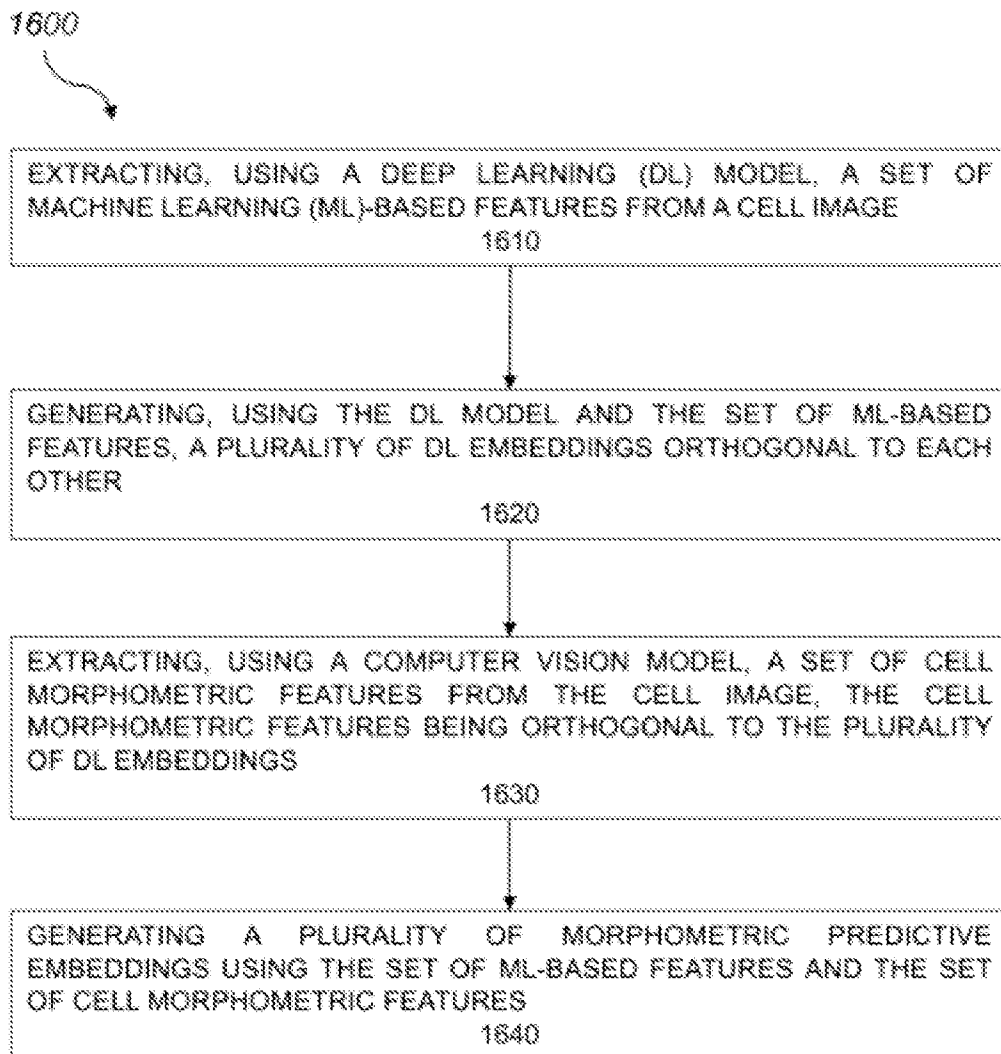


FIG. 15

**Fig. 16**

1700



EXTRACTING, USING A DEEP LEARNING (DL) MODEL AND IMAGE DATA OF A PLURALITY OF CELLS, A VECTOR FOR A CELL OF THE PLURALITY OF CELLS, THE VECTOR COMPRISING A SET OF MACHINE-LEARNING (ML)-BASED FEATURES AND A SET OF CELL MORPHOMETRIC FEATURES EXTRACTED USING A COMPUTER VISION MODEL

1710



GENERATING, USING THE DL MODEL AND USING AT LEAST THE SET OF ML-BASED FEATURES, A PLURALITY OF DL EMBEDDINGS ORTHOGONAL TO EACH OTHER

1720

Fig. 17

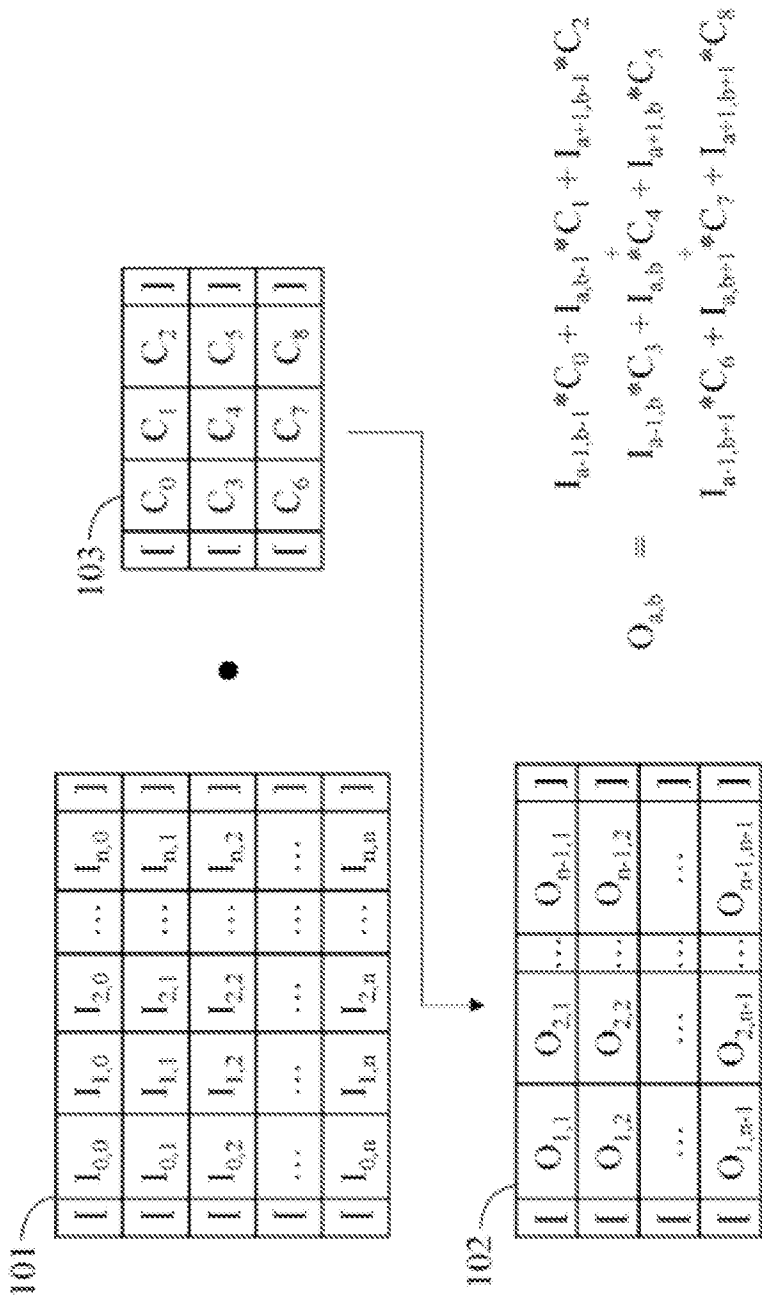


FIG. 18

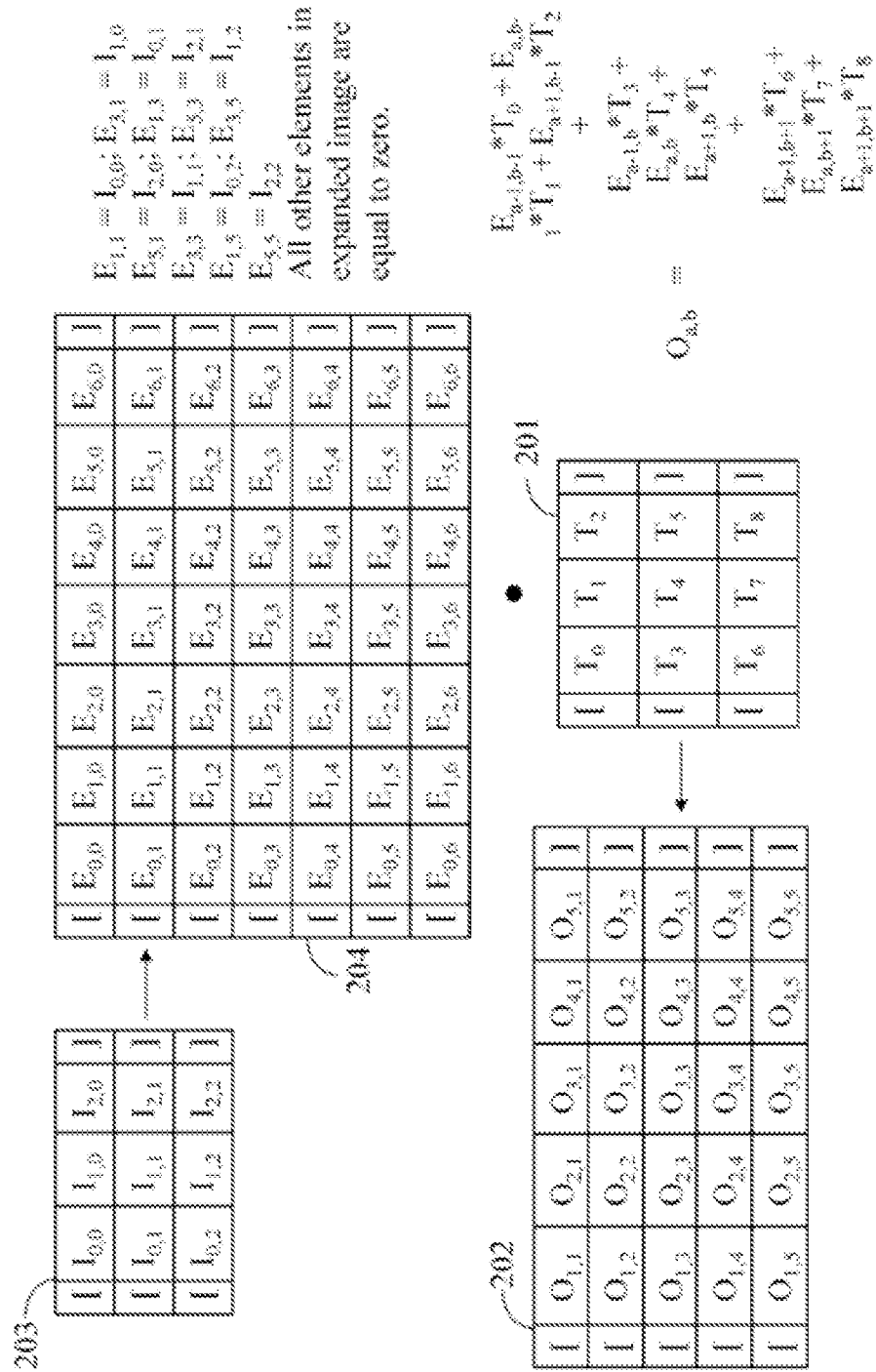


FIG. 19

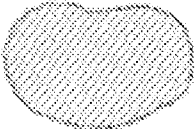
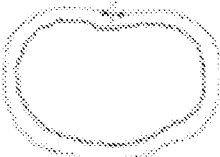
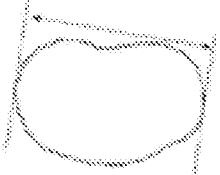
Category	Deepcell Morphometric Dimension #	Morphometric Name	Description	Schematic
Position Features	HFMMDC001	Centroid X axis (µm)	X axis position of the cell relative to the camera's field of view	
	HFMMDC002	Centroid Y axis (µm)	Y axis position of the cell relative to the camera's field of view	
Cell Shape Features	HFMMDC003	Area (µm ²)	Cell area	
	HFMMDC004	Perimeter (µm)	Length of the cell outline	
	HFMMDC005	Maximum caliper distance (µm)	Width of widest possible box around the cell	
	HFMMDC006	Minimum caliper distance (µm)	Width of narrowest possible box around the cell	
	HFMMDC007	Maximum radius (µm)	Largest radius from center of cell to cell border	
	HFMMDC008	Minimum radius (µm)	Shortest radius from center of cell to cell border	

FIG. 20A

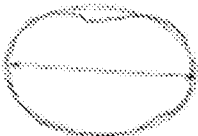
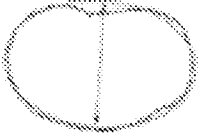
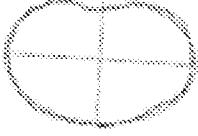
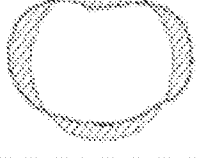
Category	Deepwell Morphometric Dimension #	Morphometric Name	Description	Schematic
	HFMDC009	Long ellipse axis (µm)	Long axis of best fit ellipse	
	HFMDC010	Short ellipse axis (µm)	Short axis of best fit ellipse	
	HFMDC011	Ellipse elongation (unitless)	Aspect ratio of best fit ellipse Metric: 0 indicates a circle, 1 indicates a line	
	HFMDC012	Ellipse similarity (unitless)	Deviation from an elliptical shape Metric: 0 indicates a perfectly elliptical shape	
	HFMDC013	Roundness (unitless)	Roundness is a measure for circularity or compactness of the shape Metric: Range is 0 to 1, where 1 is a perfect circle	
	HFMDC014	Circle similarity (unitless)	Deviation from a circular shape Metric: 0 indicates a perfectly circular shape	
	HFMDC015	Convex shape (unitless)	Ratio of the area of the convex hull of the cell to the total area of the cell	

FIG. 20B

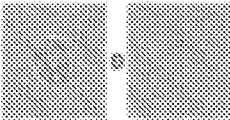
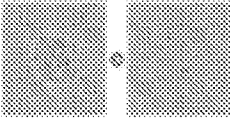
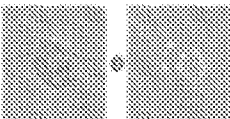
Category	Dimension #	Name	Description	Schematic
Pixel Intensity Features	HFMMDC016	Mean pixel intensity (arbitrary units)	Mean pixel grayscale value, refers to how much light a cell absorbs and/or reflects Metric range is -1 to 1	
	HFMMDC017	Standard deviation of pixel intensity	Standard deviation of pixel grayscale values gives an indication of the uniformity of pixel intensity within the cell	
	HFMMDC018	Pixel intensity 25th percentile	25th percentile of pixel grayscale values	
	HFMMDC019	Pixel intensity 75th percentile	75th percentile of pixel grayscale values	
	HFMMDC020	Positive fraction	Fraction of pixels with a grayscale value significantly above background	
	HFMMDC021	Negative fraction	Fraction of pixels with a grayscale value significantly below background	
Texture Features	HFMMDC022	Small set of connected bright pixels', integral	Sum of pixel intensities in regions identified as being small bright structures Small is described as ~8 pixels but smaller sets could be discarded	
	HFMMDC023	Small set of connected dark pixels', integral	Sum of pixel intensities in regions identified as being small dark structures Small is defined as ~8 pixels but smaller sets could be discarded	
	HFMMDC024	Large set of connected bright pixels', integral	Sum of pixel intensities in regions identified as being large bright structures Large is defined as ~32 pixels but larger sets could be discarded	

FIG. 20C

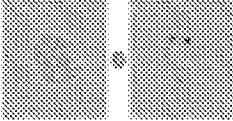
Category	Deepcell Morphometric Dimension #	Morphometric Name	Description	Schematic
	HFMMDC025	Large set of connected dark pixels*, integral	Sum of pixel intensities in regions identified as being large dark structures Large is defined as ~32 pixels but larger sets could be detected	
	HFMMDC026 - HFMMDC027	Image moments (2 moments)	Two values that describe the weighted average or distribution of the image pixel intensities within the cell in a rotation and scale-invariant manner	
	HFMMDC028 - HFMMDC037	Local binary patterns - center (10 patterns)	The 10 Local Binary Pattern (LBP) - Center features determine the texture inside the cell and describe the appearance near the center of the cell	
	HFMMDC038 - HFMMDC047	Local binary patterns - periphery (10 patterns)	The 10 Local Binary Pattern (LBP) - Periphery features determine the texture at the periphery of the cell and describe the appearance near the edge of the cell	
Focus Features	HFMMDC048	Image sharpness (arbitrary units)	A measure of how sharp or smooth the image is. Typically an out-of-focus image is less sharp than an in-focus image.	
	HFMMDC049	Image focus (µm)	Estimate of the distance of the cell to the focal plane of the microscope	

FIG. 20D

Category	Deepcell Morphometric Dimension #	Morphometric Name	Description	Schematic
	HFMMDC50	ring width (arbitrary units)	Our imaging modality creates a dark or bright ring to appear around the cell, which is larger and more intense the more out of focus the cell is. This feature estimates the width of the ring.	
	HFMMDC51	Ring intensity (arbitrary units)	Our imaging modality creates a dark or bright ring to appear around the cell, which is larger and more intense the more out of focus the cell is. This feature estimates the intensity of the ring.	

FIG. 20E

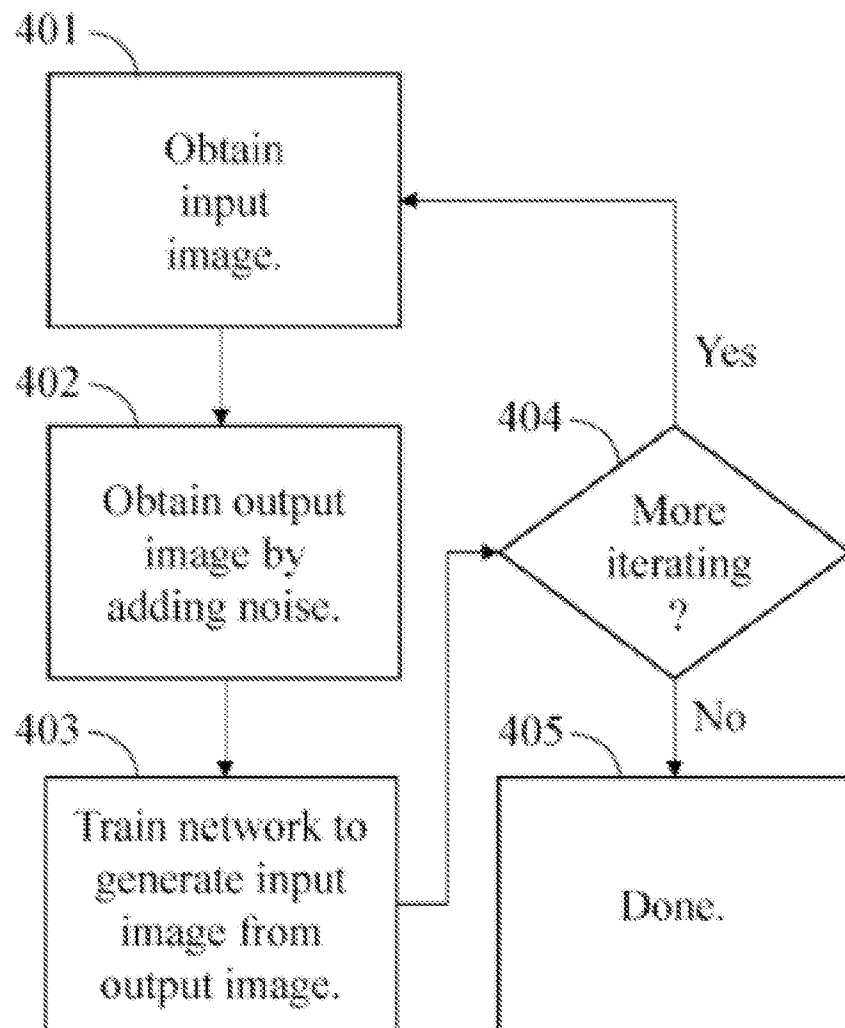


FIG. 21

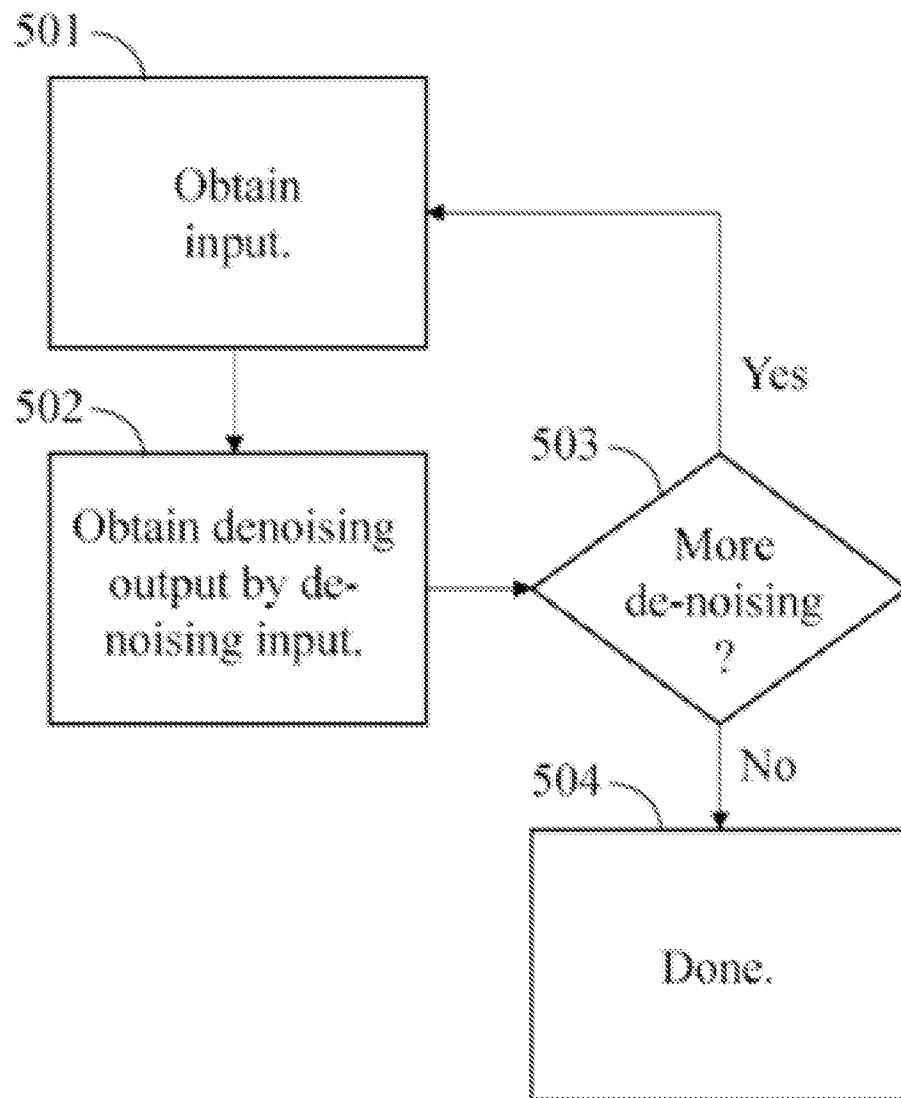


FIG. 22

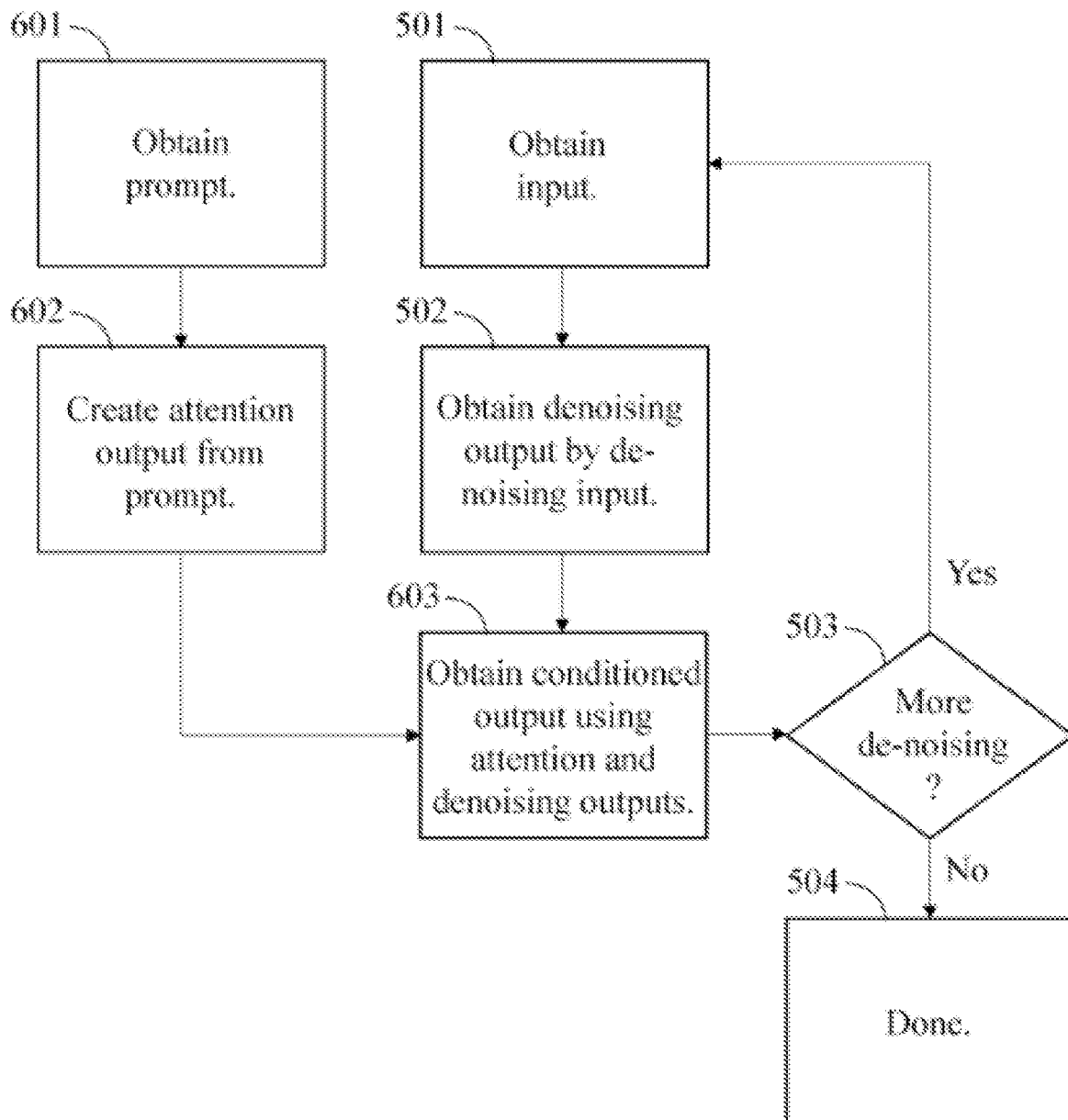


FIG. 23

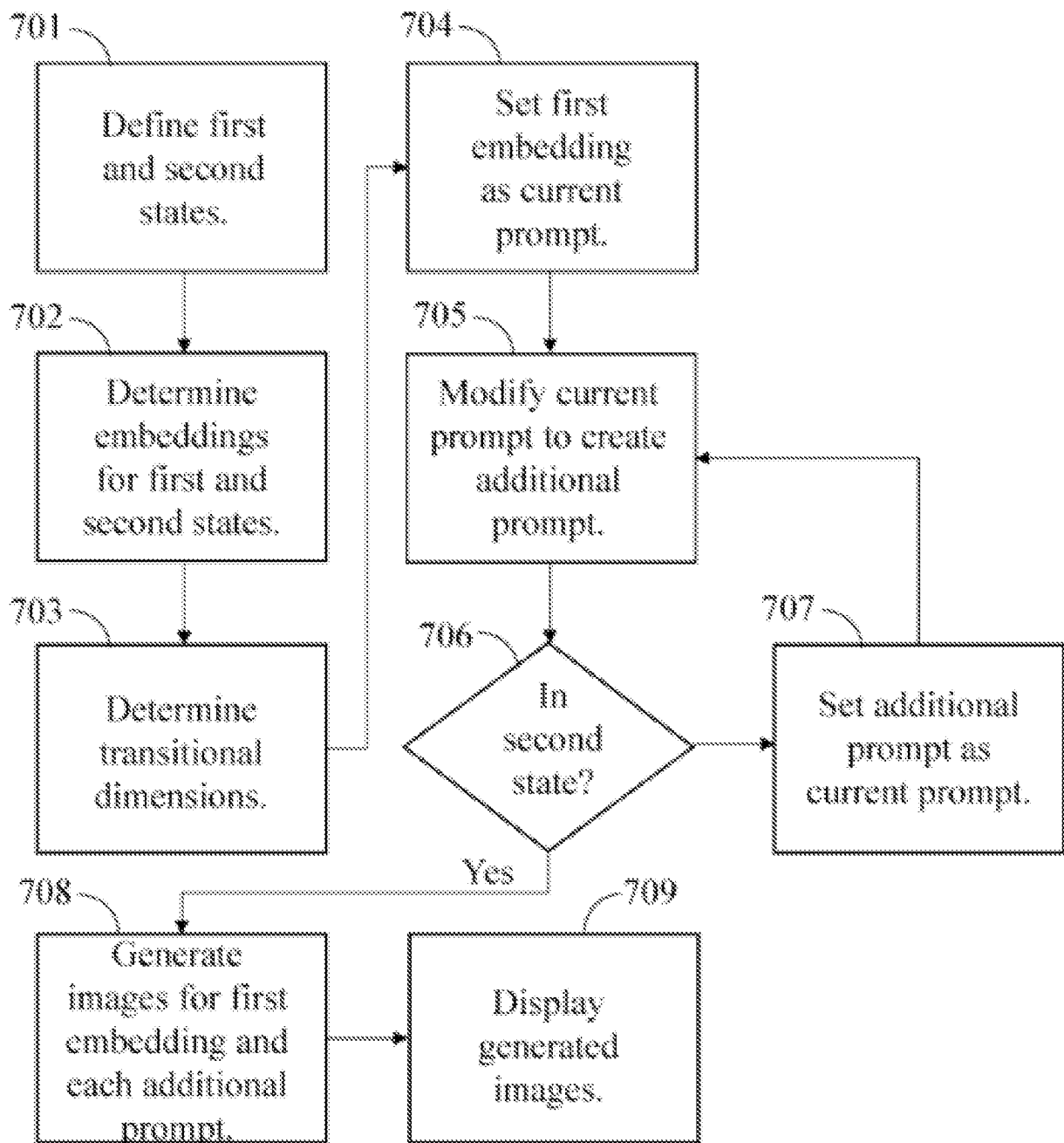


FIG. 24

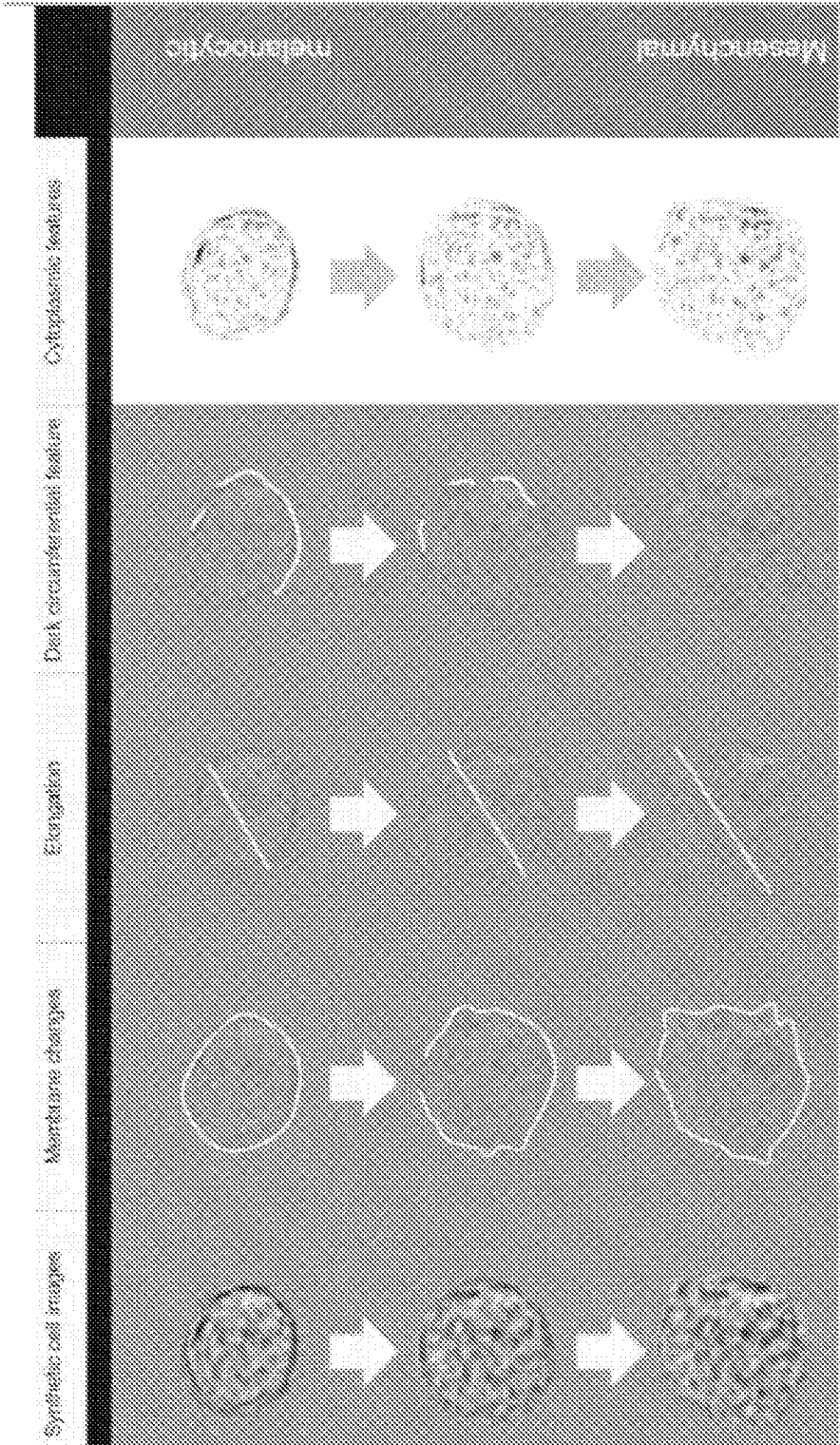


FIG. 25

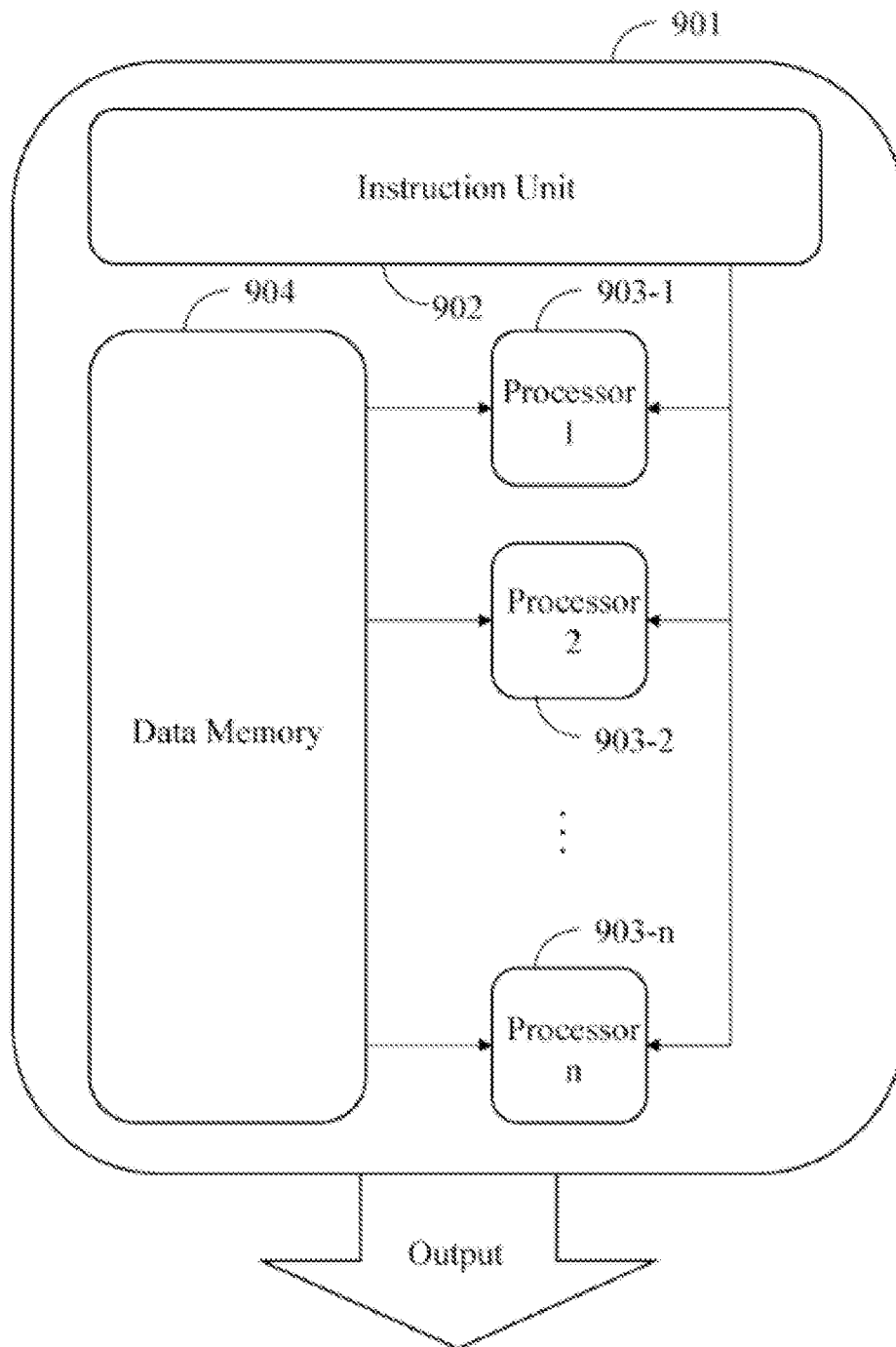


FIG. 26

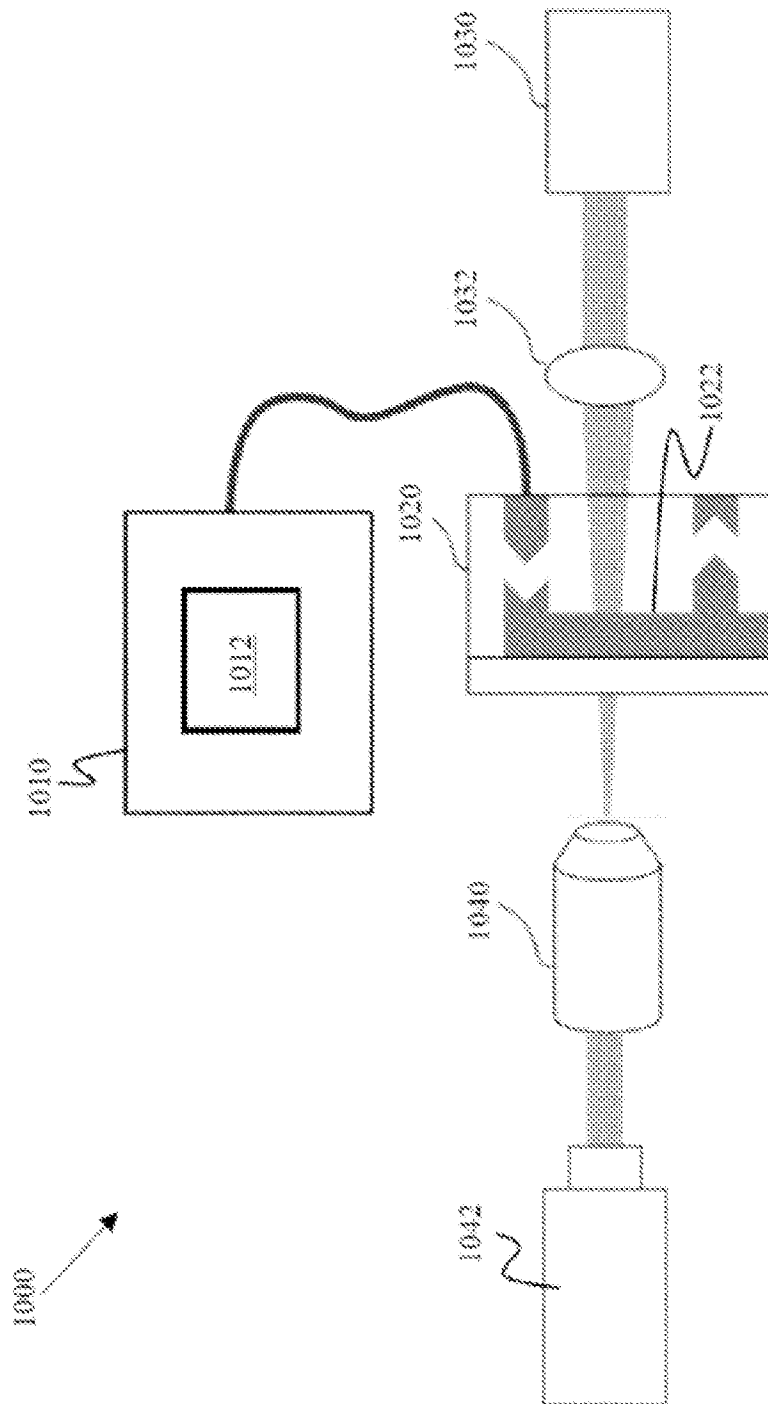


FIG. 27

INTERNATIONAL SEARCH REPORT

International application No.

PCT/US2024/053235

Box No. II Observations where certain claims were found unsearchable (Continuation of item 2 of first sheet)

This international search report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1. ☐ Claims Nos.:
because they relate to subject matter not required to be searched by this Authority, namely:
2. ☐ Claims Nos.:
because they relate to parts of the international application that do not comply with the prescribed requirements to such an extent that no meaningful international search can be carried out, specifically:
3. ☒ Claims Nos.: **55-69, 74-90**
because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

Box No. III Observations where unity of invention is lacking (Continuation of item 3 of first sheet)

This International Searching Authority found multiple inventions in this international application, as follows:

This application contains the following inventions or groups of inventions which are not so linked as to form a single general inventive concept under PCT Rule 13.1.

Group I: claims 1-47 are directed to an imaging device configured to capture images of the cell as the cell is transported through the flow channel; a processor comprising extracting machine learning (ML)-based features and cell morphometric features from the cell image; generating a plurality of morphometric predictive embeddings using the extracted ML-based features and the generated cell morphometric features; and enabling sorting of the cell using one of the generated plurality of morphometric predictive embeddings, the generated plurality of DL embeddings, and the extracted cell morphometric features.

Group II: claims 48-54 and 70-73 are directed to generating a set of prompts and a set of images; generating an image corresponding to the prompt, which is provided to a trained generative artificial intelligence model; and displaying a user interface to present the set of images.

INTERNATIONAL SEARCH REPORT

International application No.

PCT/US2024/053235

Box No. III Observations where unity of invention is lacking (Continuation of item 3 of first sheet)

1. ☐ As all required additional search fees were timely paid by the applicant, this international search report covers all searchable claims.
2. ☐ As all searchable claims could be searched without effort justifying additional fees, this Authority did not invite payment of additional fees.
3. ☐ As only some of the required additional search fees were timely paid by the applicant, this international search report covers only those claims for which fees were paid, specifically claims Nos.:
4. ☒ No required additional search fees were timely paid by the applicant. Consequently, this international search report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.: **1-47**

Remark on Protest

- ☐ The additional search fees were accompanied by the applicant's protest and, where applicable, the payment of a protest fee.
- ☐ The additional search fees were accompanied by the applicant's protest but the applicable protest fee was not paid within the time limit specified in the invitation.
- ☐ No protest accompanied the payment of additional search fees.

INTERNATIONAL SEARCH REPORT

International application No.

PCT/US2024/053235

A. CLASSIFICATION OF SUBJECT MATTER

IPC: **G06N 20/00** (2025.01); **G06V 10/40** (2025.01); G06V 10/82 (2025.01); G06V 20/69 (2025.01); G16B 20/00 (2025.01)
CPC: **G06N 20/00**; **G06V 10/40**; G06V 10/82; G06V 20/695; G16B 20/00; G06V 20/698

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

See Search History Document

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

See Search History Document

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

See Search History Document

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	WO 2022/178095 A1 (Deepcell, Inc.) 25 August 2022 (25.08.2022) entire document (especially para [0076]-[0077], [0086]-[0087], [0173]-[0180], [0300]-[0302]).	1-47
Y	US 2021/0147922 A1 (ALTIUS INSTITUTE FOR BIOMEDICAL SCIENCES) 20 May 2021 (20.05.2021) entire document (especially para [0408]-[0409], [0470])	1-47
A	US 2022/0163513 A1 (University of Washington) 26 May 2022 (26.05.2022) entire document	1-47

☐ Further documents are listed in the continuation of Box C.

☐ See patent family annex.

* Special categories of cited documents:

“A” document defining the general state of the art which is not considered to be of particular relevance

“D” document cited by the applicant in the international application

“E” earlier application or patent but published on or after the international filing date

“L” document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

“O” document referring to an oral disclosure, use, exhibition or other means

“P” document published prior to the international filing date but later than the priority date claimed

“T” later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

“X” document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

“Y” document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

“&” document member of the same patent family

Date of the actual completion of the international search

12 February 2025 (12.02.2025)

Date of mailing of the international search report

21 February 2025 (21.02.2025)

Name and mailing address of the ISA/US

**COMMISSIONER FOR PATENTS
MAIL STOP PCT, ATTN: ISA/US
P.O. Box 1450
Alexandria, VA 22313-1450
UNITED STATES OF AMERICA**

Authorized officer

KARI RODRIQUEZ

Facsimile No. **571-273-8300**

Telephone No. **PCT Help Desk: 571-272-4300**