(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
23 March 2023 (23.03.2023)

WIPO | PCT

(10) International Publication Number
WO 2023/041168 A1

(51) International Patent Classification:
G06V 10/25 (2022.01) *G06V 10/82* (2022.01)

(21) International Application Number:
PCT/EP2021/075556

(22) International Filing Date:
17 September 2021 (17.09.2021)

(25) Filing Language: English

(26) Publication Language: English

(71) Applicant: **HUAWEI TECHNOLOGIES CO., LTD.** [CN/CN]; Huawei Administration Building Bantian Longgang District, Shenzhen, Guangdong 518129 (CN).

(72) Inventor; and

(71) Applicant *(for MN only)*: **VO, Nhat** [VN/SE]; Huawei Technologies Sweden AB Skalholtsgatan 9, 16440 Kista (SE).

(72) Inventor: **XIA, Baiqiang**; Huawei Technologies Sweden AB Skalholtsgatan 9, 16440 Kista (SE).

(74) Agent: **KREUZ, Georg**; Huawei Technologies Duesseldorf GmbH Riesstr. 25, 80992 Munich (DE).

(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, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, IT, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO,

(54) **Title:** DETERMINING REGIONS OF INTEREST IN AN IMAGE

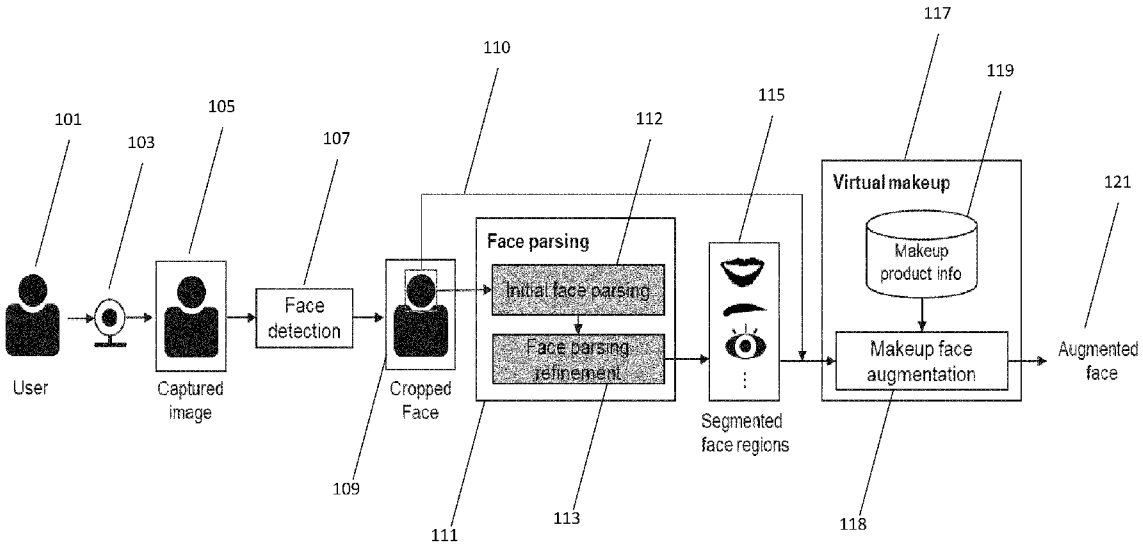

Figure 1

(57) **Abstract:** In some examples, a system for determining regions of interest in an image comprises a convolutional neural network (CNN). The CNN can comprise a down sampling pathway defining an encoder comprising a set of convolutional layers configured to output a down sampled image representation of the image, and an up sampling pathway defining a decoder configured to output classified image data for the image representing the regions of interest, the encoder configured to receive image data representing the image, and generate, using the image data and a set of kernels, output data comprising the down sampled image representation, the decoder comprising a set of layers configured to perform transposed convolutions on the down sampled image representation to generate the classified image data, wherein the decoder is further configured to receive region data representing region information for the image, the region data comprising a vector encoding one or more characteristics of at least a portion of a region of interest of

WO 2023/041168 A1

[Continued on next page]

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, GH, GM, KE, LR, LS, MW, MZ, NA, RW, 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, 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).

Published:

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

the image, and determined using the image data and a mask comprising an output of the convolutional neural network in the form of candidate classified image data.

DETERMINING REGIONS OF INTEREST IN AN IMAGE

TECHNICAL FIELD

The present disclosure relates, in general, to image segmentation. Aspects of the disclosure relate to semantic segmentation of images of faces.

BACKGROUND

A deep neural network (DNN) is an artificial neural network (ANN) with multiple layers between the input and output layers. One such DNN that is commonly applied to the analysis of visual imagery is a Convolutional Neural Network (CNN). For example, CNNs are commonly used to detect faces in images. A detected face can be segmented such that pixel-wise labels are assigned to each identified semantic component, e.g., hair, eyes, nose, mouth, etc., in an image. That is, each pixel can be marked with a semantic label thereby resulting in a semantically rich face map which can be used for a variety of high-level applications ranging from virtual 'try-on' systems for, e.g., spectacles or make-up and so on, to augmented reality systems.

Face segmentation or parsing processes rely heavily on training in which, given a set of input face images and their target segmented face maps, a supervised model can be optimized to predict the best-segmented face maps possible from input face images. Generally, face segmentation models may not work well on particular face images that do not appear in the training set or that suffer from low image quality. This can be because the supervised models that are employed tend to be overfitted to the training data or simply cannot predict a correct output when presented with unseen samples. For applications requiring highly accurate face maps, inaccurate face segmentation results can lead to poor performance of subsequent modules which otherwise rely on a precise semantic labelling for the image in question. For example, an imprecise semantic labeling applied to an image passed to an augmented reality module of a system can result in a sub-optimal user experience as a result of incorrect or inaccurate image augmentation.

SUMMARY

An objective of the present disclosure is to enable generation of accurate face maps in which semantic components have been segmented. The foregoing and other objectives are achieved by the features of the independent claims. Further implementation forms are apparent from the dependent claims, the description and the Figures.

A first aspect of the present disclosure provides a system for determining regions of interest in an image, the system comprising a convolutional neural network comprising a down sampling pathway defining an encoder comprising a set of convolutional layers configured to output a down sampled image representation of the image, and an up sampling pathway defining a decoder configured to output classified image data for the image representing the regions of interest, the encoder configured to receive image data representing the image, and generate, using the image data and a set of kernels, output data comprising the down sampled image representation, the decoder comprising a set of layers configured to perform transposed convolutions on the down sampled image representation to generate the classified image data, wherein the decoder is further configured to receive region data representing region information for the image, the region data comprising a vector encoding one or more characteristics of at least a portion of a region of interest of the image, and determined using the image data and a mask comprising an output of the convolutional neural network in the form of candidate classified image data.

Accordingly, a highly accurate face map can be generated by refinement of face parsing results in which region information is leveraged in order to guide the refinement. As such, inaccurate region maps and region information can be used as guidance for a face parsing refinement model. The region information can be extracted from a target face region, and embedded into the refinement model. Typical image segmentation models are designed for general applications and generally lack application-specific guidance. Therefore, over-segmentation and under-segmentation of regions in face parsing results are problematic. According to aspects of the present disclosure, region maps and region information can be used to refine the face maps, whilst training can be performed using limited annotated data.

An image of a person, such as a user of a mobile device (user equipment) can be processed in order to detect one or more faces in that image. The portion(s) of the image comprising the detected face(s) can be parsed in order to semantically segment the portion(s), whereby to

enable regions of the portion(s) to be determined which correspond to facial features or components, such as eyes, nose, mouth and so on. Characteristics or attributes of these features or components are, in an example, extracted in order to provide data that can be used to refine a face map.

In an implementation of the first aspect, a region information extractor can be configured to extract, for a region of interest, data representing one or more image characteristics comprising: data representing a colour histogram, data representing texture and data representing shape using the image data and the mask, and for each image characteristic, generate a characteristic or feature vector using the extracted data. As noted above, the region information can be used as guidance for a face parsing refinement model.

In an example, the region information extractor can concatenate each characteristic or feature vector to generate the vector encoding one or more characteristics of at least a portion of the region of interest of the image. Thus, each of the data embodying a feature vector can be used to generate a single vector representing region data for an image patch for example. The region information extractor can rescale the vector encoding one or more characteristics to the same dimension as the output of the encoder.

A quality estimation module can compare an output of the decoder with a predetermined threshold representing an end condition for the system, and based on the comparison, provide final classified image data. The quality estimation module can activate a refinement iteration in the event that the comparison indicates that the output of the decoder does not meet the predetermined threshold.

According to a second aspect of the present disclosure, there is provided a method for determining regions of interest in an image using a convolutional neural network, the method comprising generating region data representing region information for the image, the region data comprising a vector encoding one or more characteristics of at least a portion of a region of interest of the image and generated using image data representing the image and a mask comprising a previous output of the convolutional neural network in the form of candidate classified image data, generating output data comprising a down sampled representation of the image using an encoder of the convolutional neural network, and generating classified image data for the image representing the regions of interest using an up sampling pathway defining a decoder of the convolutional neural network by weighting the output data according to a

predefined profile using the region data, and performing transposed convolutions on the so weighted output data.

In an implementation of the second aspect, the method can further comprise extracting, for a region of interest, data representing one or more image characteristics comprising: data representing a colour histogram, data representing texture and data representing shape using the image data and the mask, and for each image characteristic, generating a characteristic vector using the extracted data. The method can further comprise concatenating each characteristic vector to generate the vector encoding one or more characteristics of at least a portion of the region of interest of the image. The method can further comprise rescaling the vector encoding one or more characteristics to the same dimension as the output of the encoder. The method can further comprise comparing an output of the decoder with a predetermined threshold representing an end condition for the system, and based on the comparison, providing final classified image data. The method can further comprise activating a refinement iteration in the event that the comparison indicates that the output of the decoder does not meet the predetermined threshold. The method can further comprise generating a ground truth score by comparing the output of the convolutional neural network with a ground truth map of the image, and generating the predetermined threshold using the ground truth score. The method can further comprise augmenting the regions of interest with augmentation data, whereby to generate an augmented image.

According to a third aspect of the present disclosure, there is provided user equipment comprising a memory encoded with instructions for determining regions of interest in an image generated using an imaging module, the instructions executable by a processor of the user equipment, whereby to cause the user equipment to generate region data representing region information for the image, the region data comprising a vector encoding one or more characteristics of at least a portion of a region of interest of the image and generated using image data representing the image and a mask comprising a previous output of a convolutional neural network in the form of candidate classified image data, generate output data comprising a down sampled representation of the image using an encoder of the convolutional neural network, and generate classified image data for the image representing the regions of interest using an up sampling pathway defining a decoder of the convolutional neural network by weighting the output data according to a predefined profile using the region data, and performing transposed convolutions on the so weighted output data.

In an implementation of the third aspect, the user equipment can comprise a region information extractor to extract, for a region of interest, data representing one or more image characteristics comprising: data representing a colour histogram, data representing texture and data representing shape using the image data and the mask, and for each image characteristic, generate a characteristic vector using the extracted data.

In an example, the region information extractor can concatenate each characteristic vector to generate the vector encoding one or more characteristics of at least a portion of the region of interest of the image. The region information extractor can rescale the vector encoding one or more characteristics to the same dimension as the output of the encoder.

The user equipment can further comprise a quality estimation module configured to compare an output of the decoder with a predetermined threshold representing an end condition for the system, and based on the comparison, provide final classified image data. The quality estimation module can activate a refinement iteration in the event that the comparison indicates that the output of the decoder does not meet the predetermined threshold.

These and other aspects of the invention will be apparent from the embodiment(s) described below.

BRIEF DESCRIPTION OF THE DRAWINGS

In order that the present invention may be more readily understood, embodiments of the invention will now be described, by way of example, with reference to the accompanying drawings, in which:

Figure 1 is a schematic representation of a system according to an example;

Figure 2 is a schematic representation of a system according to an example;

Figure 3 is a schematic representation of the generation of training maps according to an example;

Figure 4 is a schematic representation of the generation of region data according to an example;

Figure 5 is a schematic representation of a system according to an example;

Figure 6 is a flowchart depicting a refinement process according to an example;

Figure 7 is a schematic representation of a system according to an example; and

Figure 8 is a schematic representation of a machine according to an example.

DETAILED DESCRIPTION

Example embodiments are described below in sufficient detail to enable those of ordinary skill in the art to embody and implement the systems and processes herein described. It is important to understand that embodiments can be provided in many alternate forms and should not be construed as limited to the examples set forth herein.

Accordingly, while embodiments can be modified in various ways and take on various alternative forms, specific embodiments thereof are shown in the drawings and described in detail below as examples. There is no intent to limit to the particular forms disclosed. On the contrary, all modifications, equivalents, and alternatives falling within the scope of the appended claims should be included. Elements of the example embodiments are consistently denoted by the same reference numerals throughout the drawings and detailed description where appropriate.

The terminology used herein to describe embodiments is not intended to limit the scope. The articles “a,” “an,” and “the” are singular in that they have a single referent, however the use of the singular form in the present document should not preclude the presence of more than one referent. In other words, elements referred to in the singular can number one or more, unless the context clearly indicates otherwise. It will be further understood that the terms “comprises,” “comprising,” “includes,” and/or “including,” when used herein, specify the presence of stated features, items, steps, operations, elements, and/or components, but do not preclude the presence or addition of one or more other features, items, steps, operations, elements, components, and/or groups thereof.

Unless otherwise defined, all terms (including technical and scientific terms) used herein are to be interpreted as is customary in the art. It will be further understood that terms in common usage should also be interpreted as is customary in the relevant art and not in an idealized or overly formal sense unless expressly so defined herein.

Face segmentation or parsing is a process in which every pixel of an image of a face is classified into a category of facial components. Detecting different facial components is of great interest for, e.g., augmented reality (AR) applications such as facial image beautification and facial image editing. For example, having classified the lip area in an image of a face, virtual lipstick can be applied by colorizing the region, thereby enabling someone to see the effects of the different colours without having to physically apply different colored lipsticks. This is but one example and there are numerous other applications that either rely on or that can leverage a classified image of a face. For example, semantic segmentation of an image can be usefully applied in the fields of autonomous vehicles, bio-medical imaging, geo sensing, agriculture and so on.

In any application, accurate semantic segmentation is key to enable a satisfying end user experience, whilst ensuring that computational complexity is minimised, particularly in the case where applications may be implemented on platforms with otherwise limited resources, such as mobile platforms in the form of user equipment such as smart phones and the like.

According to an example, an image segmentation process is provided in which low quality face maps can be refined from limited training data in order to achieve highly accurate face maps. Face parsing results can be iteratively refined to generate a highly precise face map. That is, in an example, for an input face image and an inaccurate face map, a refinement process can be employed to improve the face maps iteratively. Inaccurate face maps for training can be obtained directly from the result of a previous refinement and/or by augmenting ground truth face maps. In an example, a refinement process leverages region information to guide refinement. Region information can capture or comprise, e.g., information representing or defining the shape, and/or color, and/or texture characteristics of a face region. In an example, a refinement process can also predict quality scores of improved face maps, such as recall, precision and so on that can be used to select the most refined face maps from each iteration.

An image segmentation process according to an example can use a CNN comprising an encoder-decoder structure. For example, a symmetric encoder-decoder fully convolutional network can be used in which the encoder downsamples an input image and the decoder upsamples a corresponding feature map in order to reconstruct an output, which can be in the form of a high-resolution image (typically of the same size as input image) in which each pixel is classified to a particular class, thereby forming a pixel level image classification. In an example, upsampling can be performed by transposed convolutional operations

(deconvolution). Thus, a CNN architecture according to an example comprises two pathways. The first pathway, referred to as the encoder contracts the input and is used to capture the context in an image. The encoder can comprise, e.g., a set of convolutional and max pooling layers. The second pathway, referred to as the decoder, comprises a symmetric expanding path configured to enable localization using transposed convolutions.

According to an example, the decoder can receive region data representing region information for the image under consideration. This region data can comprise data representing one or more characteristics of at least a portion of a region of interest of the image, such as the shape, and/or color, and/or texture characteristics of a face region for example. The region data can comprise a vector encoding these characteristics for the image and can be determined using the image data and a mask comprising an output of the CNN in the form of candidate classified image data. That is, region data can be used to refine a face map generated by the system.

Figure 1 is a schematic representation of a system according to an example. In the example of figure 1, a user 101 uses an image capture device 103, such as a camera provided as part of a user equipment for example, to generate a captured image 105 of themselves. It will be appreciated that the image capture device 103 may be a stand alone appliance and/or that the captured image 105 may be of a person other than the user 101. A face detection module 107 implementing a face detector can be used to detect the face in the captured image. The output of the face detection module 107 is an image 109 that includes data representing the location of the detected face 110. This may typically comprise a bounding box for example configured to encompass or contain the detected face. Such face detectors are well known and will not be described in any further detail.

In block 111 face parsing or segmentation is performed using the CNN as briefly described above. As part of the process in block 111, an initial face parsing 112 is performed. The initial face parsing 112 is followed by a refinement process in block 113 that generates a set of segmented face regions 115. In combination with the detected face 110, the segmented face regions 115 can be, in the example of figure 1, used by a virtual makeup application 117 in which the user 101 may select products stored in a repository of product information 119 to be applied to the detected face 110 at the appropriate positions corresponding to the segmented face regions 115 as part of a makeup face augmentation process 118 to generate an augmented face representation 121. For example, user 101 may select a lipstick of a certain colour. Accordingly, the makeup face augmentation process 118 can colourise the appropriate

segmented face region 115 (i.e., in this case the lips). The colourised lips can be overlaid on the detected face 110 at the position of the lips to form the augmented face representation 121.

Figure 2 is a schematic representation of a system according to an example. In the example of figure 2, the system 200 is configured to determine regions of interest in an image using a CNN 201. CNN 201 comprises a down sampling pathway defining an encoder 203. The encoder 203 comprises a set of convolutional layers configured to output a down sampled image representation 207 of the image 105. CNN 201 further comprises an up sampling pathway defining a decoder 205 configured to output classified image data 209 for the image 105 representing the regions of interest. In the example of figure 2, one such region of interest, in the form of the mouth (i.e., lips) is depicted. In order to generate the image data 209, the encoder can receive image data 211 representing the image and generate, using the image data 211 and a set of kernels, output data comprising the down sampled image representation 207. The decoder 205, which comprises a set of layers configured to perform transposed convolutions on the down sampled image representation 207, can generate the classified image data 209. In an example, the decoder 205 is further configured to receive region data 213 representing region information for the image. The region data can comprise a vector encoding one or more characteristics of at least a portion of a region of interest of the image. The region data 213 can be determined using the image data 211 and a mask 215 comprising an output of the convolutional neural network in the form of candidate classified image data.

With reference to figure 2, system 200 therefore defines an image segmentation network that comprises convolutional layers to transform an input image to embedded features and deconvolutional layers to transform the embedded features back into a target face map. Initially, a face image and an empty face map can be fed into the system, which is trained to predict initial face maps by applying, e.g., a backpropagation algorithm.

Initial face maps can be iteratively refined by incorporating the region information into the image segmentation network backbone. The refinement process generates a higher accuracy face map from an initial (inaccurate) face map and its corresponding image with guidance from the extracted region information. The improved face map can then be used as the input for the next refinement iteration. During training input face maps, Map_{t-1}, can be generated and collected from two sources: (1) directly from the result of previous iterations, and (2) augmented face maps derived from the ground-truth map 217. For example, random deformations can be applied to the original ground truth map 217 to simulate incorrect

predictions of segmented face maps. Deformations can comprise at least one of, e.g., morphological operations, object shape shrinking/expansion, and spatial transformations.

Figure 3 is a schematic representation of the generation of training maps according to an example. As noted above, face maps for training 301 can be generated and collected from the result of previous iterations (or predictions) 303, and/or augmented face maps derived from the ground-truth map 217. For example, random deformations 302 can be applied to the original ground truth map 217 to simulate incorrect predictions of segmented face maps. Deformations can comprise at least one of, e.g., morphological operations, object shape shrinking/expansion, and spatial transformations.

Figure 4 is a schematic representation of the generation of region data according to an example. As noted above, region data can comprise information that captures at least one of the shape, color, and texture characteristics of at least a portion of a face region. In an example, it can be extracted as feature vectors within a region map and different types of region information can be extracted, such as the color histogram, region texture, region shape, and so on. With reference to figure 4, an image patch 401 of an image and a corresponding mask 403 representing a segmented map for the image patch are depicted, which, in this example comprises a mouth. The mask 403 is used to demarcate the areas on the image patch 401 from which region information is to be extracted. So, for example, the mask 403 can comprise a binary map in which the lips from the image patch 401 are white and the remainder of the mask 403 is black. Accordingly, in this example, white portions of the mask 403 represent portions of the image patch 401 from which region information will be extracted. In the example of figure 4, at least one of data 405 representing a color histogram, data 407 representing texture and data 409 representing shape of the feature 401 can be extracted. Data 405 representing a color histogram comprises a measure of the distribution in colors of the feature. Different color systems can be used, such as RGB, HSV, YCbCr, etc. In an example, the color histogram of the region can be represented as one dimension feature vector. Data 407 representing texture can be obtained using any one of different texture extraction processes, such as Local Binary Pattern (LBP) which comprises a texture operator which labels the pixels of an image by thresholding the neighborhood of each pixel and considers the result as a binary number, a Gabor filter which comprises a linear filter used to determine the presence of specific frequency content in the image portion in specific directions in a localized region around the point or region of analysis, or a histogram of oriented gradients (HOG) in which counts of occurrences of gradient orientation in localized portions of an image are obtained. Data 409 representing

shape can comprise data representing features such as contour curvature, area function, centroid distance function, and so on.

Each of the data 405, 407, 409 comprise respective feature vectors 411. The feature vectors 411 from each type of region information can be concatenated into a single vector 413 to create the region data. The region data 413 therefore comprises a vector encoding one or more characteristics of at least a portion of a region of interest of an image patch 401, which is determined using the image data and a mask 403 comprising an output of the convolutional neural network in the form of candidate classified image data.

So, with reference to the image patch 401 for example, information extracted from the image patch in view of the map 403 could comprise a vector defining a colour histogram representing the colour of the lips of the mouth. Other information could comprise a vector in which the curvature of the lips has been characterised in a vector for example. Furthermore, the texture of the lips will differ from that of the surrounding face and teeth in the image patch, which is again information that can be embodied in a corresponding feature vector. Such information can then be formed into the vector 413 as described above and used to further refine the output of the CNN 201 by enabling, e.g., weighting to be applied to certain areas of an image to be segmented.

Figure 5 is a schematic representation of a system according to an example. A generator branch 501 is constructed to map the region data comprising the information vector 413 to the 2D embedded feature space of the CNN. In an example, the generator branch 501 of the CNN 201 can be composed of a series of a strided 2D convolutional transpose layers (deconvolutional layers) 503, each paired with a 2D batch norm layer and a rectified linear unit (ReLU) activation. The output 505 of the branch generator 501 is rescaled (507) to the size of the network backbone's embedded features (207) output from the encoder 203 to form a resized output 508. The output 508 can be integrated (511) into the network backbone's embedded features 207 to form an input 513 for the decoder 205. For example, the output 207 of the encoder 203 can be weighted and/or scaled using the output 508. In an example, this can lend more weight to regions of interest in the deconvolution pathway of the CNN 201 in order to enable an improved map to be generated. Accordingly, concatenated features are passed to the deconvolutional block 205 to generate the refined face map 209. The generator branch can be also trained end-to-end in the same manner as the network backbone branch.

The use of inaccurate face maps in training teaches the model how to improve the face maps. In an inference stage, with the iterative refinement strategy described above, the quality of refined face maps improves with each iteration. Furthermore, the training map generation solves the problem of limited data since, in theory, unlimited low-quality maps can be generated for training the refinement model. Since traditional image segmentation models are designed for general applications and lack application-specific guidance they can produce over- or under-segmentation of regions in face parsing results. According to an example, using region maps and region information a better model can be defined to refine face maps. Furthermore, limited annotated data can be used for training without the need to add new (annotated) training data. In an inference stage, missing areas can be automatically complemented and redundant ones can be removed based on the previous region mask and the region information. This is advantageous when the inference stage is faced with a hard example that would otherwise provide a sub-optimal result.

Figure 6 is a flowchart depicting a refinement process according to an example. In the example of figure 6 a quality estimation process 601 is configured to estimate the quality scores of refined face maps, as will be described in more detail below. In block 603 the CNN 201 estimates the initial face map 209. The initial face map is used to predict a quality score in block 601. If, in block 605, the quality score is below a predetermined threshold value, the network predicts a refined face map in block 607 and determines a quality score in block 601 again. In the event that a quality score is above the predetermined threshold value, the face map can be considered a suitable output (block 609) for an application such as, e.g., virtual make-up try - on. In an example, a threshold value can be set as a hyper-parameter and the iteration of figure 6 ends when: (1) the quality score threshold is met, or (2) a max number of iterations is reached. Given extracted features from the last layers of the refinement model, the quality estimation process can be used to regress the quality scores of a predicted map. The quality scores may represent any relevant metric, e.g., IoU, Precision, Recall, etc. In training, the quality estimation module can be trained together with the face map refinement head or alternatively by updating mask prediction and quality prediction. The ground truth of quality scores can be determined directly from the refined Map_t 209 and the ground truth map 217.

Figure 7 is a schematic representation of a system according to an example. In the example of figure 7, a quality estimation process 601 is depicted in more detail. The quality estimation process 601 can be implemented using a series of convolutional layers 701, each paired with a 2D batch norm layer and a ReLU activation. The feature maps from the final convolutional

layers of the decoder 205 of the CNN 201 can be converted to 1D feature vectors by fully connected layers. The target is to regress the quality scores of refined Map_t 209 based on the metrics estimated from Map_t and the ground truth map 217. Different image segmentation metrics can be computed. For example, recall, precision, or intersection over union (IoU) can be used to estimate the quality of refined maps, depending on the application. Thus, a feature map 209 output from the CNN 201 can be compared against the ground truth 217 using one of the methods outlined above in order to generate a measure representing a ground truth score 703. This can be compared against a predicted quality score 705 in order to determine whether the quality of the feature map 209 satisfies the predetermined threshold value described with reference to figure 6.

In an inference stage, there are two options that can be implemented to provide a stopping condition and to select a suitable refined face map. For example, a stopping condition can be based on the metric threshold: that is, the refinement iterations will be stopped as long as one, several, or all of the quality scores surpass a threshold or a list of thresholds as noted above. The recently predicted map is then selected as the final result. Alternatively, a stopping condition can be based on a maximum number of iterations: that is, the refinement iterations can be stopped when the number of refinement iterations meets or exceeds a threshold number. In this case, the refined map with the highest quality scores is selected as the final result. Quality scores can therefore be used as a criteria to select a suitable map, depending on the application.

According to an example, the application of multiple quality scores provides a system that is more effective in deciding the quality of refined face maps. Furthermore, since different quality scores represent different quality aspects of a face map, they can increase flexibility in selecting suitable face maps for an application.

The system employs the inaccurate region maps and region information as guidance for the face parsing refinement model. Different types of region information are extracted from the target face region, and this information is embedded into the refinement model in the form of 2D embedded feature maps. In training, the inaccurate region maps come from two sources: (1) directly from the result of previous training iterations and (2) from randomly deformed ground-truth maps. In inference, the model first obtains the initial face parsing results. Then the model iteratively refines the face maps by using the region information and previous refinements.

The quality estimation module predicts the quality/accuracy of the face parsing results. It automatically regresses different metrics based on the estimation between the refined face map

and the ground truth. Finally, a threshold on the quality prediction or a max number of iteration is used to decide the termination of the iterative refinement process automatically.

By using region map and region information, a better model to refine face maps can be provided. The model can be trained from limited annotated data without adding new training data. In inference, the model can automatically complement missing areas and remove redundant ones based on the previous region mask and the region information. Therefore, more precise face maps can be obtained. A refinement can automatically decide when to stop based on the estimated quality scores. The quality scores can be used as the criteria to select suitable face maps for each application.

Examples in the present disclosure can be provided as methods, systems or machine-readable instructions, such as any combination of software, hardware, firmware or the like. Such machine-readable instructions may be included on a computer readable storage medium (including but not limited to disc storage, CD-ROM, optical storage, etc.) having computer readable program codes therein or thereon.

The present disclosure is described with reference to flow charts and/or block diagrams of the method, devices and systems according to examples of the present disclosure. Although the flow diagrams described above show a specific order of execution, the order of execution may differ from that which is depicted. Blocks described in relation to one flow chart may be combined with those of another flow chart. In some examples, some blocks of the flow diagrams may not be necessary and/or additional blocks may be added. It shall be understood that each flow and/or block in the flow charts and/or block diagrams, as well as combinations of the flows and/or diagrams in the flow charts and/or block diagrams can be realized by machine readable instructions.

The machine-readable instructions may, for example, be executed by a machine such as a general-purpose computer, user equipment such as a smart device, e.g., a smart phone, a special purpose computer, an embedded processor or processors of other programmable data processing devices to realize the functions described in the description and diagrams. In particular, a processor or processing apparatus may execute the machine-readable instructions. Thus, modules of apparatus (for example, a module implementing an encoder or decoder of the CNN) may be implemented by a processor executing machine readable instructions stored in a memory, or a processor operating in accordance with instructions embedded in logic circuitry.

The term 'processor' is to be interpreted broadly to include a CPU, processing unit, ASIC, logic unit, or programmable gate set etc. The methods and modules may all be performed by a single processor or divided amongst several processors.

Such machine-readable instructions may also be stored in a computer readable storage that can guide the computer or other programmable data processing devices to operate in a specific mode. For example, the instructions may be provided on a non-transitory computer readable storage medium encoded with instructions, executable by a processor.

Figure 8 is a schematic representation of a machine according to an example. The machine 800 can be, e.g., part of a system or apparatus, user equipment (or part thereof). The machine 800 comprises a CNN 801. The machine 800 comprises a processor 803, and a memory 805 to store instructions 807, executable by the processor 803. The machine comprises a storage 809 that can be used to store captured images, detected faces, cropped faces, segmented face regions, image patches, feature maps, product information and so on as described above with reference to figures 1 to 7 for example.

The instructions 807, executable by the processor 803, can cause the machine 800 to generate region data representing region information for a captured image or a portion thereof, the region data comprising a vector encoding one or more characteristics of at least a portion of a region of interest of the image and generated using image data representing the image and a mask comprising a previous output of a convolutional neural network in the form of candidate classified image data, generate output data comprising a down sampled representation of the image using an encoder of the convolutional neural network, and generate classified image data for the image representing the regions of interest using an up sampling pathway defining a decoder of the convolutional neural network by weighting the output data according to a predefined profile using the region data, and performing transposed convolutions on the so weighted output data.

Accordingly, the machine 800 can implement a method for determining regions of interest in an image using a convolutional neural network. Such machine-readable instructions may also be loaded onto a computer or other programmable data processing devices, so that the computer or other programmable data processing devices perform a series of operations to produce computer-implemented processing, thus the instructions executed on the computer or other

programmable devices provide an operation for realizing functions specified by flow(s) in the flow charts and/or block(s) in the block diagrams.

Further, the teachings herein may be implemented in the form of a computer or software product, such as a non-transitory machine-readable storage medium, the computer software or product being stored in a storage medium and comprising a plurality of instructions, e.g., machine readable instructions, for making a computer device implement the methods recited in the examples of the present disclosure.

In some examples, some methods can be performed in a cloud-computing or network-based environment. Cloud-computing environments may provide various services and applications via the Internet. These cloud-based services (e.g., software as a service, platform as a service, infrastructure as a service, etc.) may be accessible through a web browser or other remote interface of the user equipment for example. Various functions described herein may be provided through a remote desktop environment or any other cloud-based computing environment.

While various embodiments have been described and/or illustrated herein in the context of fully functional computing systems, one or more of these exemplary embodiments may be distributed as a program product in a variety of forms, regardless of the particular type of computer-readable-storage media used to actually carry out the distribution. The embodiments disclosed herein may also be implemented using software modules that perform certain tasks. These software modules may include script, batch, or other executable files that may be stored on a computer-readable storage medium or in a computing system. In some embodiments, these software modules may configure a computing system to perform one or more of the exemplary embodiments disclosed herein. In addition, one or more of the modules described herein may transform data, physical devices, and/or representations of physical devices from one form to another.

The preceding description has been provided to enable others skilled in the art to best utilize various aspects of the exemplary embodiments disclosed herein. This exemplary description is not intended to be exhaustive or to be limited to any precise form disclosed. Many modifications and variations are possible without departing from the spirit and scope of the instant disclosure. The embodiments disclosed herein should be considered in all respects illustrative and not

restrictive. Reference should be made to the appended claims and their equivalents in determining the scope of the instant disclosure.

CLAIMS

1. A system for determining regions of interest in an image, the system comprising a convolutional neural network comprising:

a down sampling pathway defining an encoder comprising a set of convolutional layers configured to output a down sampled image representation of the image; and

an up sampling pathway defining a decoder configured to output classified image data for the image representing the regions of interest, the encoder configured to:

receive image data representing the image; and

generate, using the image data and a set of kernels, output data comprising the down sampled image representation;

the decoder comprising a set of layers configured to perform transposed convolutions on the down sampled image representation to generate the classified image data, wherein the decoder is further configured to receive region data representing region information for the image, the region data comprising a vector encoding one or more characteristics of at least a portion of a region of interest of the image, and determined using the image data and a mask comprising an output of the convolutional neural network in the form of candidate classified image data.

2. The system as claimed in claim 1, further comprising a region information extractor configured to:

extract, for a region of interest, data representing one or more image characteristics comprising: data representing a colour histogram, data representing texture and data representing shape using the image data and the mask; and

for each image characteristic, generate a characteristic vector using the extracted data.

3. The system as claimed in claim 2, wherein the region information extractor is configured to concatenate each characteristic vector to generate the vector encoding one or more characteristics of at least a portion of the region of interest of the image.

4. The system as claimed in claim 2 or 3, wherein the region information extractor is further configured to:

rescale the vector encoding one or more characteristics to the same dimension as the output of the encoder.

5. The system as claimed in any preceding claim, further comprising a quality estimation module configured to:

compare an output of the decoder with a predetermined threshold representing an end condition for the system; and

based on the comparison, provide final classified image data.

6. The system as claimed in claim 5, wherein the quality estimation module is configured to:

activate a refinement iteration in the event that the comparison indicates that the output of the decoder does not meet the predetermined threshold.

7. A method for determining regions of interest in an image using a convolutional neural network, the method comprising:

generating region data representing region information for the image, the region data comprising a vector encoding one or more characteristics of at least a portion of a region of interest of the image and generated using image data representing the image and a mask comprising a previous output of the convolutional neural network in the form of candidate classified image data;

generating output data comprising a down sampled representation of the image using an encoder of the convolutional neural network; and

generating classified image data for the image representing the regions of interest using an up sampling pathway defining a decoder of the convolutional neural network by:

weighting the output data according to a predefined profile using the region data; and

performing transposed convolutions on the so weighted output data.

8. The method as claimed in claim 7, further comprising:

extracting, for a region of interest, data representing one or more image characteristics comprising: data representing a colour histogram, data representing texture and data representing shape using the image data and the mask; and

for each image characteristic, generating a characteristic vector using the extracted data.

9. The method as claimed in claim 8, further comprising:

concatenating each characteristic vector to generate the vector encoding one or more characteristics of at least a portion of the region of interest of the image.

10. The method as claimed in claim 8 or 9, further comprising:

rescaling the vector encoding one or more characteristics to the same dimension as the output of the encoder.

11. The method as claimed in any of claims 8 to 10, further comprising:

comparing an output of the decoder with a predetermined threshold representing an end condition for the system; and

based on the comparison, providing final classified image data.

12. The method as claimed in claim 11, further comprising:

activating a refinement iteration in the event that the comparison indicates that the output of the decoder does not meet the predetermined threshold.

13. The method as claimed in claim 12, further comprising:

generating a ground truth score by comparing the output of the convolutional neural network with a ground truth map of the image; and

generating the predetermined threshold using the ground truth score.

14. The method as claimed in any of claims 8 to 13, further comprising:

augmenting the regions of interest with augmentation data, whereby to generate an augmented image.

15. User equipment comprising a memory encoded with instructions for determining regions of interest in an image generated using an imaging module, the instructions executable by a processor of the user equipment, whereby to cause the user equipment to:

generate region data representing region information for the image, the region data comprising a vector encoding one or more characteristics of at least a portion of a region of interest of the image and generated using image data representing the image and a mask comprising a previous output of a convolutional neural network in the form of candidate classified image data;

generate output data comprising a down sampled representation of the image using an encoder of the convolutional neural network; and

generate classified image data for the image representing the regions of interest using an up sampling pathway defining a decoder of the convolutional neural network by:

weighting the output data according to a predefined profile using the region data; and

performing transposed convolutions on the so weighted output data.

16. The user equipment as claimed in claim 15, further comprising a region information extractor configured to:

extract, for a region of interest, data representing one or more image characteristics comprising: data representing a colour histogram, data representing texture and data representing shape using the image data and the mask; and

for each image characteristic, generate a characteristic vector using the extracted data.

17. The user equipment as claimed in claim 16, wherein the region information extractor is configured to concatenate each characteristic vector to generate the vector encoding one or more characteristics of at least a portion of the region of interest of the image.

18. The user equipment as claimed in claim 16 or 17, wherein the region information extractor is further configured to:

rescale the vector encoding one or more characteristics to the same dimension as the output of the encoder.

19. The user equipment as claimed in any of claims 15 to 18, further comprising a quality estimation module configured to:

compare an output of the decoder with a predetermined threshold representing an end condition for the system; and

based on the comparison, provide final classified image data.

20. The user equipment as claimed in claim 19, wherein the quality estimation module is configured to:

activate a refinement iteration in the event that the comparison indicates that the output of the decoder does not meet the predetermined threshold.

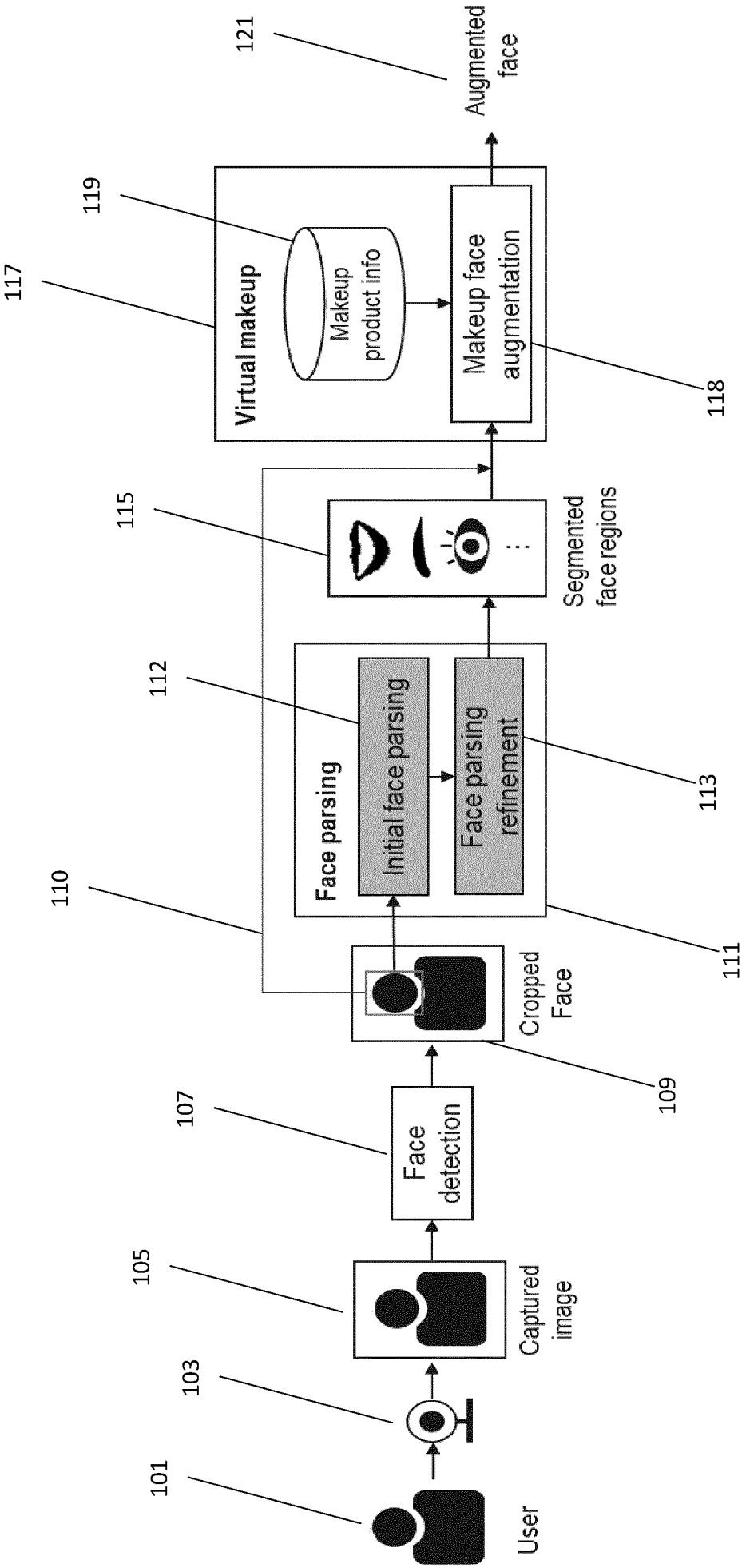
101
User
103
105
Captured image
107
Face detection
109
Cropped Face
110
111
112
Face parsing
Initial face parsing
Face parsing refinement
113
115
Segmented face regions
117
Virtual makeup
119
Makeup product info
Makeup face augmentation
118
121
Augmented face

Figure 1

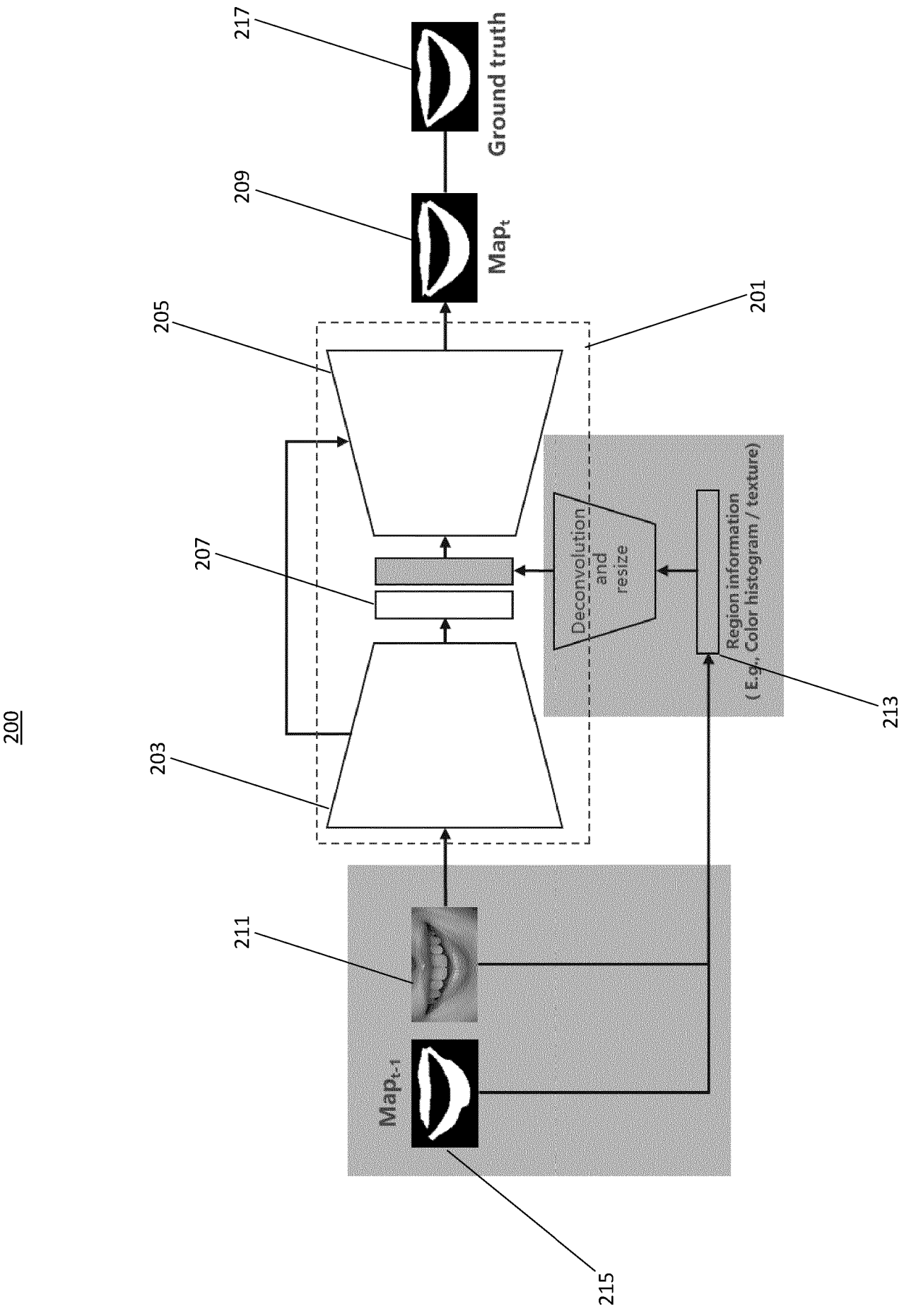
200
201
203
205
207
209
211
213
215
217
Ground truth
Map$_t$
Deconvolution and resize
Region information
(E.g., Color histogram / texture)
Map$_{t-1}$

Figure 2

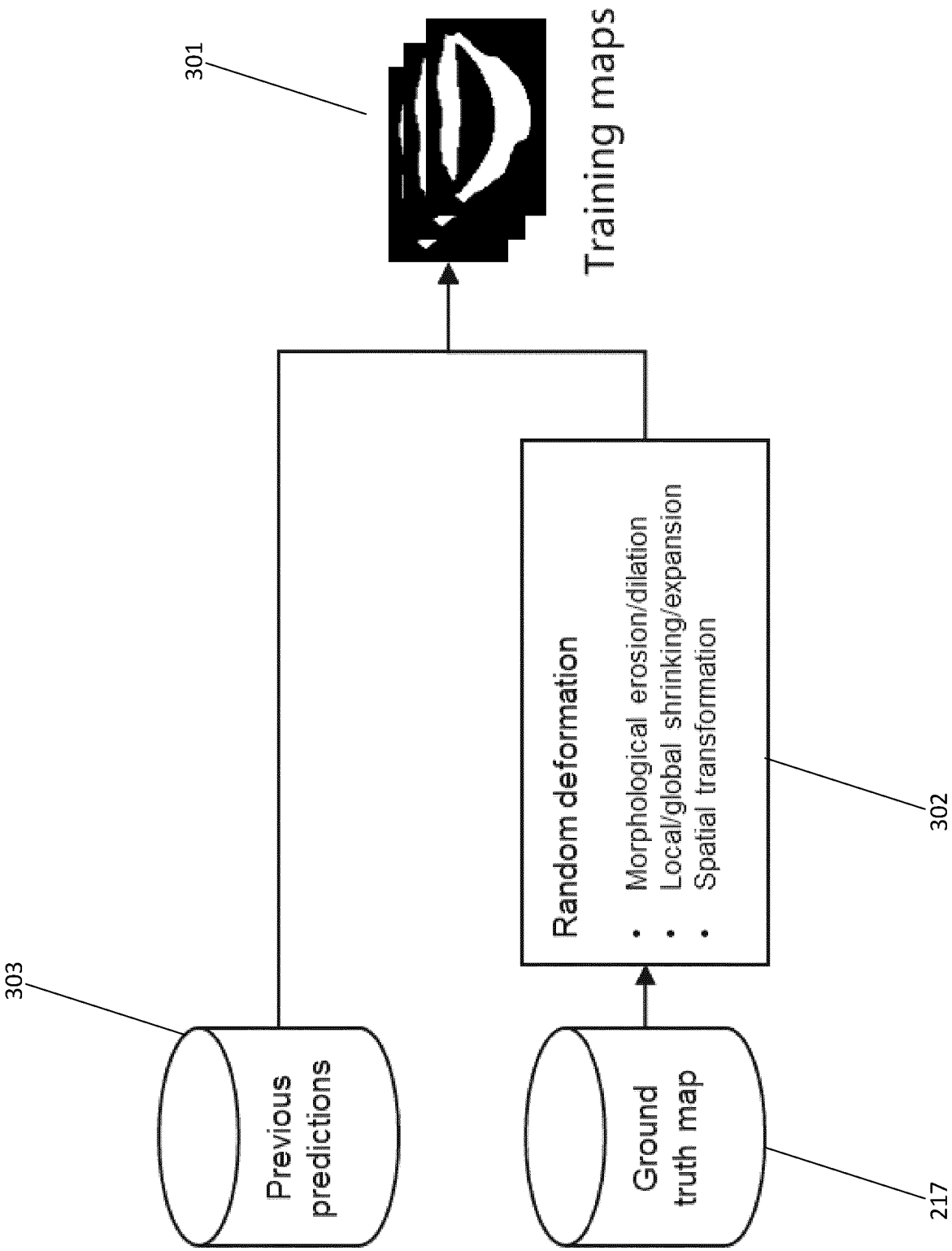
301
Training maps
Random deformation
• Morphological erosion/dilation
• Local/global shrinking/expansion
• Spatial transformation
302
303
Previous predictions
Ground truth map
217

Figure 3

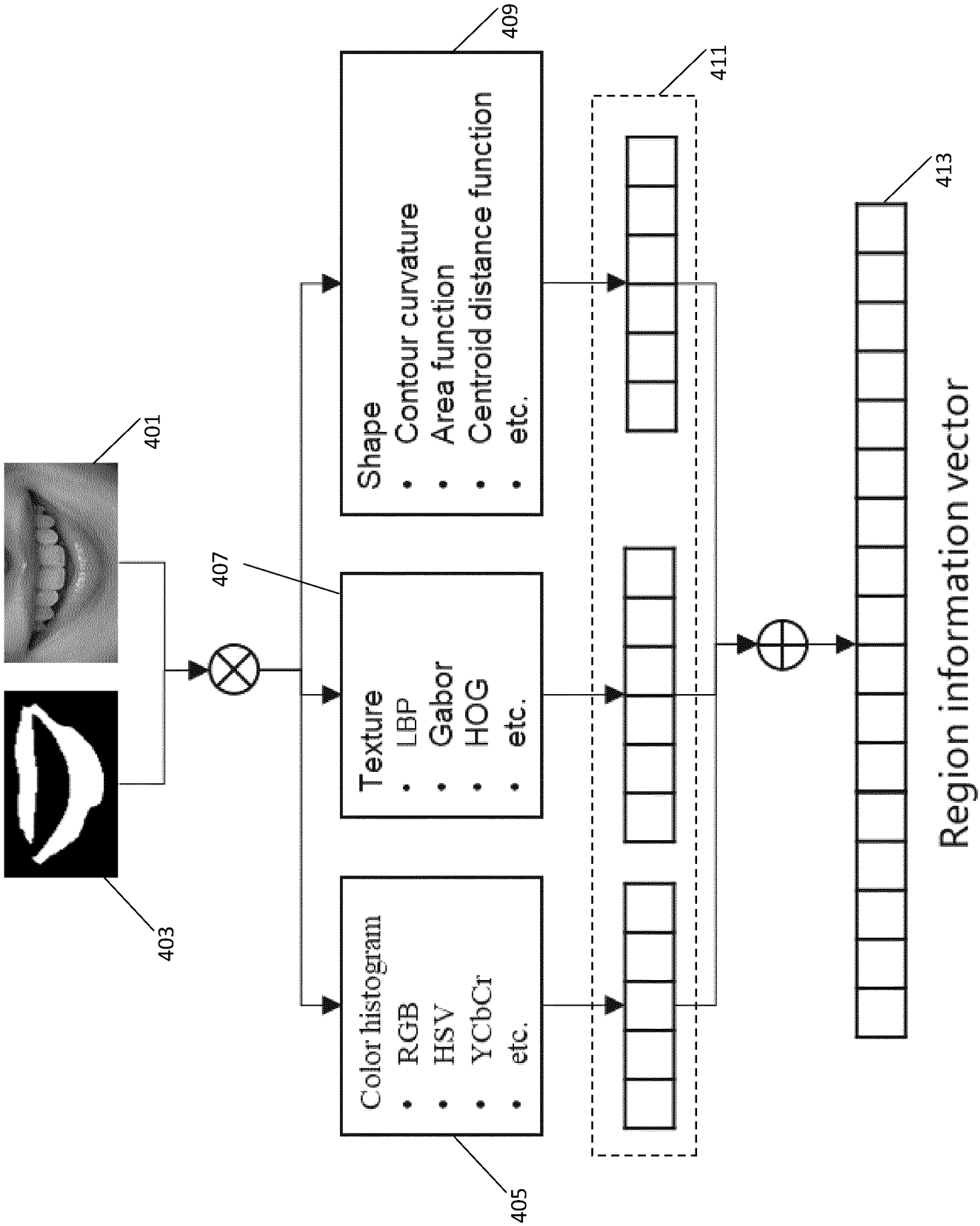
401
403
407
409
Shape
• Contour curvature
• Area function
• Centroid distance function
• etc.
411
Texture
• LBP
• Gabor
• HOG
• etc.
Color histogram
• RGB
• HSV
• YCbCr
• etc.
405
413
Region information vector

Figure 4

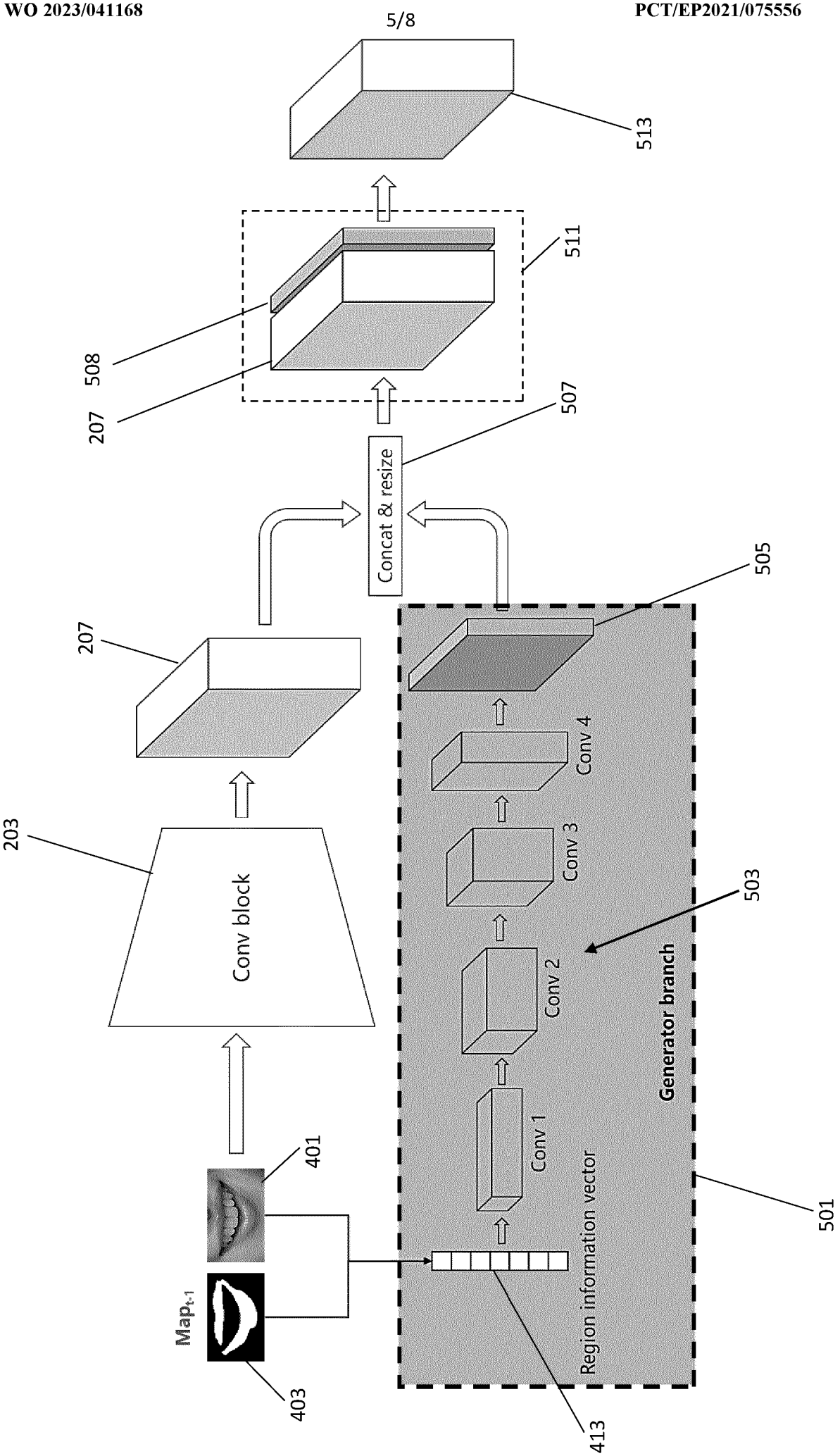
513
511
508
207
507
Concat & resize
505
207
203
Conv block
Conv 4
Conv 3
Conv 2
Conv 1
503
Generator branch
Region information vector
501
401
Map_{t-1}
403
413

Figure 5

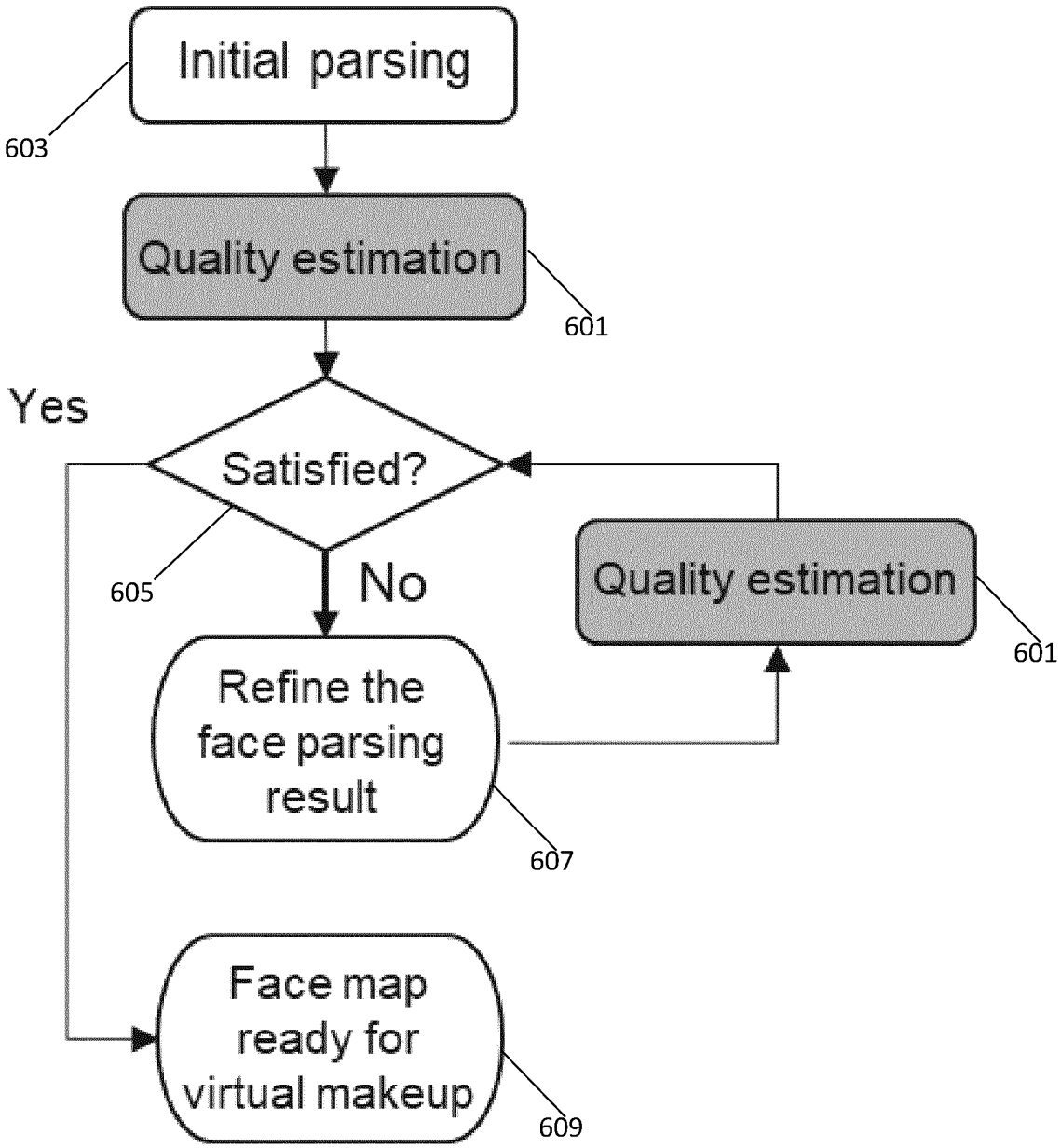
Initial parsing
603
Quality estimation
601
Yes
Satisfied?
605
No
Quality estimation
601
Refine the face parsing result
607
Face map ready for virtual makeup
609

Figure 6

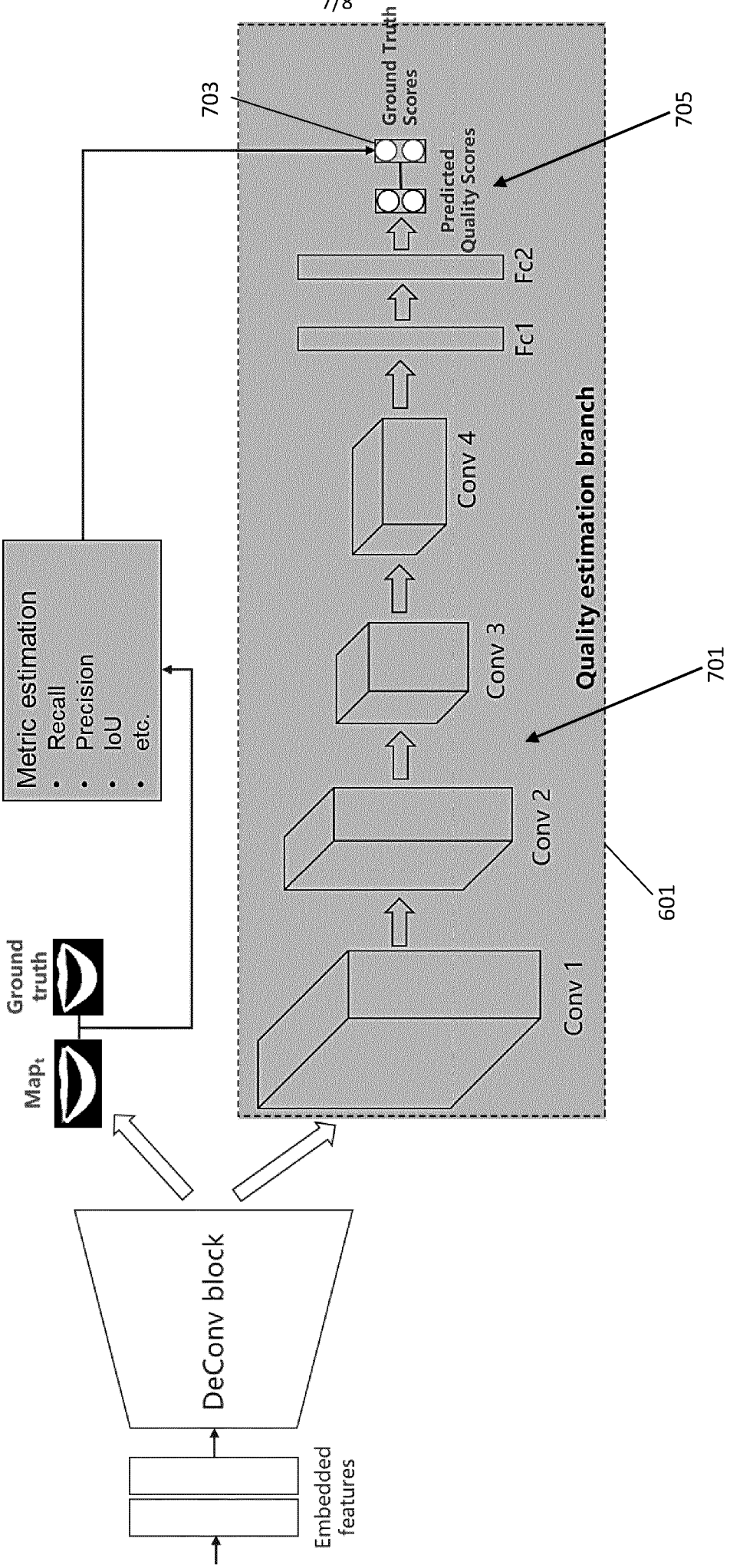
Embedded features
DeConv block
Map$_t$
Ground truth
Metric estimation
• Recall
• Precision
• IoU
• etc.
Quality estimation branch
Conv 1
Conv 2
Conv 3
Conv 4
Fc1
Fc2
Predicted Quality Scores
Ground Truth Scores
601
701
703
705

Figure 7

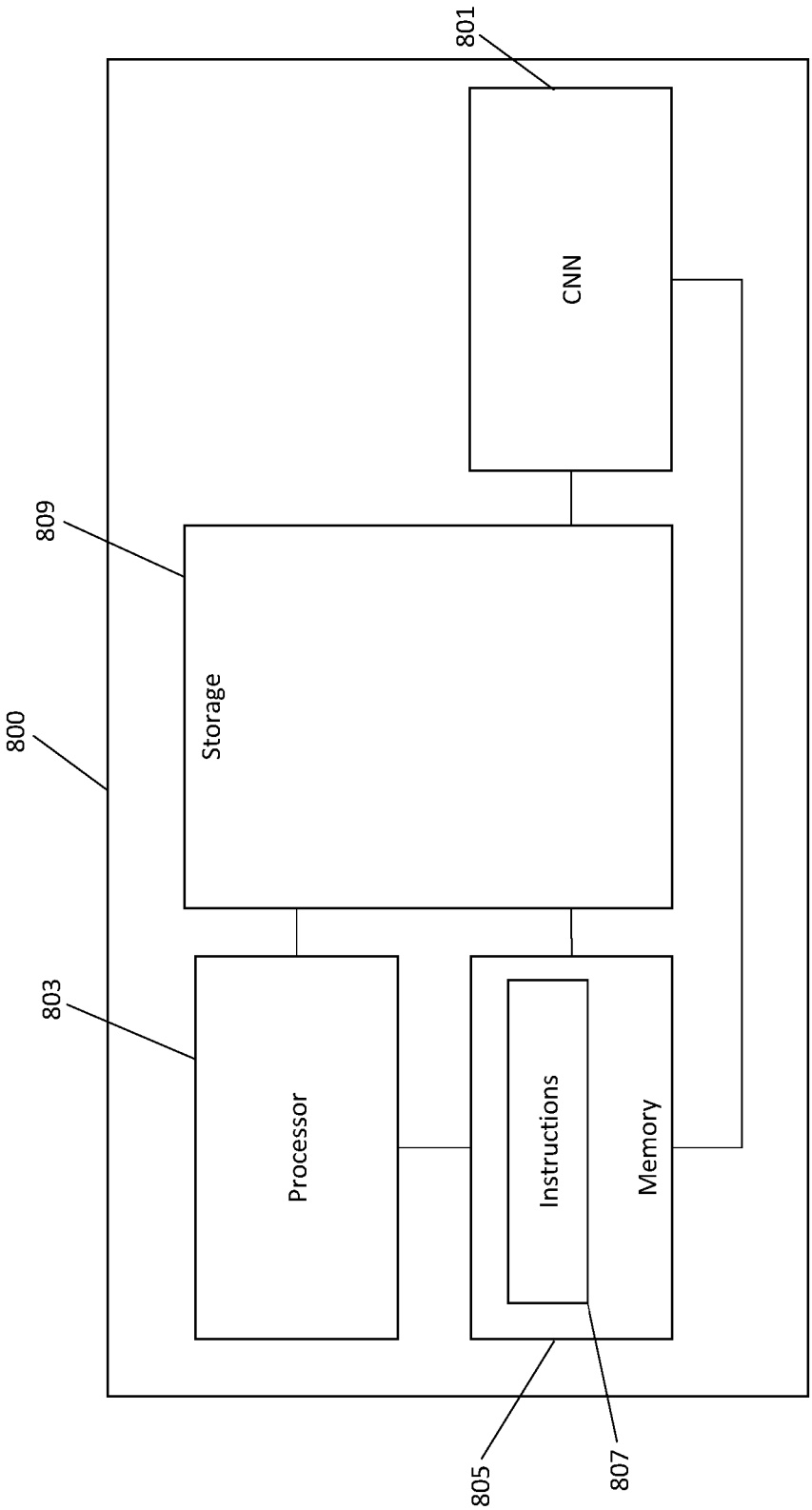
801
CNN
809
Storage
800
803
Processor
Instructions
Memory
805
807

Figure 8

INTERNATIONAL SEARCH REPORT

International application No
PCT/EP2021/075556

A. CLASSIFICATION OF SUBJECT MATTER
INV. G06V10/25 G06V10/82
ADD.

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)
G06V

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

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	OMAR ELHARROUSS ET AL: "An encoder-decoder-based method for COVID-19 lung infection segmentation", ARXIV.ORG, [Online] 4 July 2020 (2020-07-04), XP081713488, Retrieved from the Internet: URL:https://arxiv.org/pdf/2007.00861.pdf> [retrieved on 2020-07-04] Sections 3 and 4; figures 1, 3 ----- -/--	1-20

[x] 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
"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	Date of mailing of the international search report
3 June 2022	14/06/2022

Name and mailing address of the ISA/	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Craciun, Paula

2

Form PCT/ISA/210 (second sheet) (April 2005)

INTERNATIONAL SEARCH REPORT

International application No
PCT/EP2021/075556

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	LE TRUC ET AL: "REDN: A Recursive Encoder-Decoder Network for Edge Detection", IEEE ACCESS, IEEE, USA, vol. 8, 12 May 2020 (2020-05-12), pages 90153-90164, XP011790081, DOI: 10.1109/ACCESS.2020.2994160 [retrieved on 2020-05-20] Section III; figure 1	1,7,15
A	XU NING ET AL: "Deep Image Matting", 2017 IEEE CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION (CVPR), IEEE COMPUTER SOCIETY, US, 21 July 2017 (2017-07-21), pages 311-320, XP033249367, ISSN: 1063-6919, DOI: 10.1109/CVPR.2017.41 [retrieved on 2017-11-06] Section 4; figure 3	1-20
A	WANG LEZI ET AL: "A Coupled Encoder-Decoder Network for Joint Face Detection and Landmark Localization", 2017 12TH IEEE INTERNATIONAL CONFERENCE ON AUTOMATIC FACE & GESTURE RECOGNITION (FG 2017), IEEE, 30 May 2017 (2017-05-30), pages 251-257, XP033109666, DOI: 10.1109/FG.2017.40 [retrieved on 2017-06-28] Section III; figures 2, 3	1-20

2