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

(19) **World Intellectual Property Organization**
International Bureau

(43) **International Publication Date**
19 November 2020 (19.11.2020)

WIPO | PCT

(10) **International Publication Number**
WO 2020/232069 A1

(51) **International Patent Classification:**
G06T 7/70 (2017.01)

(21) **International Application Number:**
PCT/US2020/032595

(22) **International Filing Date:**
13 May 2020 (13.05.2020)

(25) **Filing Language:** English

(26) **Publication Language:** English

(30) **Priority Data:**
62/848,358 15 May 2019 (15.05.2019) US

(71) **Applicant: NORTHEASTERN UNIVERSITY** [US/US]; c/o The Center for Research Innovation, 360 Huntington Avenue, Boston, MA 02115 (US).

(72) **Inventors: FU, Yun**; 11 Cunningham Road, Wellesley, MA 02481 (US). **JIANG, Songyao**; 12 Valley St., Apt. 440, Everett, MA 02149 (US).

(74) **Agent: SOLOMON, Mark, B.** et al.; Hamilton, Brook, Smith & Reynolds, P.C., 530 Virginia Rd, P.O. Box 9133, Concord, MA 01742-9133 (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, 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, 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, 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,

(54) **Title:** VIDEO 2D MULTI-PERSON POSE ESTIMATION USING MULTI-FRAME REFINEMENT AND OPTIMIZATION

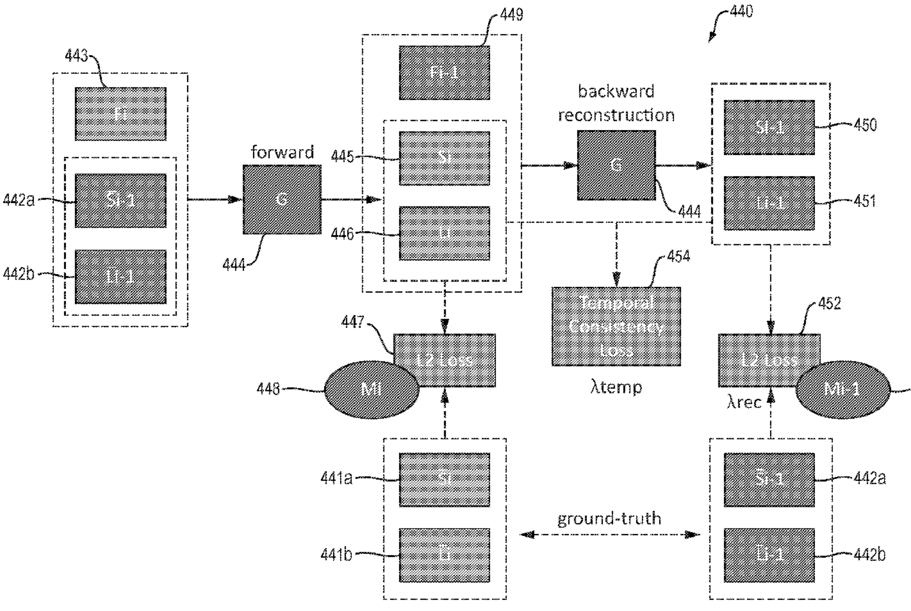

FIG. 4

(57) **Abstract:** Embodiments provide functionality for identifying joints and limbs in frames of video that use indications of joints and limbs from a previous frame. One such embodiment processes a current frame of video to determine initial predictions of joint and limb locations in the current frame. In turn, indications of the joint and limb locations in the current frame are generated by refining the initial predictions of the joint and limb locations based on indications of respective joint and limb locations from a previous frame. Embodiments provide results that are insensitive to occlusions and results that have less shaking and vibration.

[Continued on next page]

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).

Declarations under Rule 4.17:

— *of inventorship (Rule 4.17(iv))*

Published:

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

Video 2D Multi-Person Pose Estimation Using Multi-Frame Refinement and Optimization

RELATED APPLICATION

[0001] This application claims the benefit of U.S. Provisional Application No. 62/848,358, filed on May 15, 2019. The entire teachings of the above application are incorporated herein by reference.

BACKGROUND

[0002] Pose estimation, i.e., locating body parts in images, has been a computer vision task of increasing importance. Similarly, locating body parts in video, and locating body parts for multiple people in video, has become increasingly desired.

SUMMARY

[0003] While techniques exist for pose estimation and multi-person pose estimation in video, existing methods are inadequate. Many existing methods combine two separate models. These existing methods do pose estimation on each frame, track the estimated results, and, then, after performing the pose estimation, correct the results using temporal information contained in videos. This makes existing methods computationally complicated and limits the running speed of the existing methods.

[0004] Embodiments provide a novel deep learning model particularly designed and optimized for video pose estimation, which inherently takes a pose estimation result of previous frames as input to refine a new pose estimation of a current frame. Embodiments track estimated poses and make a model, i.e., a trained neural network, insensitive to occlusions. Moreover, embodiments of present invention apply a backward reconstruction loop and temporal consistency to an objective function to alleviate inconsistent estimation between adjacent frames. This significantly mitigates shaking and vibration phenomena of estimated pose skeletons in video pose estimation.

[0005] An example embodiment is directed to a method of identifying joints and limbs in a current frame of video. Such an example embodiment, first, processes the current frame of video to determine initial predictions of joint and limb locations in the current frame. In turn, indications of the joint and limb locations in the current frame are generated by refining the initial predictions of the joint and limb locations based on indications of respective joint and limb locations from a previous frame.

[0006] Another embodiment generates an indication of pose for at least one object based upon the indications of the joint and limb locations in the current frame. Embodiments may be used to identify limbs and joints of any type of object. For example, in an embodiment, the indications of the joint and limb locations in the current frame correspond to joints and limbs of at least one of: a human, animal, machine, and robot, amongst other examples. According to an embodiment, the indication of joint locations in the current frame indicates a probability of a joint at each location in the current frame and the indication of limb locations in the current frame indicates a probability of a limb at each location in the current frame. In an example embodiment, the previous frame is adjacent in time to the current frame in the video.

[0007] In an embodiment, generating the indications of the joint and limb locations in the current frame comprises processing the initial prediction of joint locations in the current frame and the indications of joint locations from the previous frame with a first deep convolutional neural network to generate the indication of joint locations in the current frame. Further, in such an embodiment, an initial prediction of limb locations in the current frame and the indications of limb locations from the previous frame are processed with a second deep convolutional neural network to generate the indication of limb locations in the current frame.

[0008] Another embodiment processes the current frame of video to determine an initial prediction of limb orientation at each initial prediction of limb location in the current frame. Further, such an embodiment generates an indication of limb orientation in the current frame by refining the initial prediction of limb orientation at each initial prediction of limb location in the current frame using indications of limb orientations from the previous frame.

[0009] Another embodiment is directed to a computer system for identifying joints and limbs in a current frame of video. The computer system includes a processor and a memory with computer code instructions stored thereon. In such an embodiment, the processor and the memory, with the computer code instructions, are configured to cause the system to identify joints and limbs according to any embodiment described herein.

[0010] Yet another embodiment is directed to a computer program product for identifying joints and limbs in a current frame of video. The computer program product comprises one or more non-transitory computer-readable storage devices and program instructions stored on at least one of the one or more storage devices. The program instructions, when loaded and executed by a processor, cause an apparatus associated with the processor to identify joints and limbs in a frame of video as described herein.

[0011] An embodiment is directed to a method of training a neural network to identify joints and limbs in a current frame of video. Such a method embodiment performs forward and backward optimization between adjacent frames of video to refine joint location prediction results and limb location prediction results of a neural network. In turn, the neural network is updated based on the refined joint location prediction results and the refined limb location prediction results.

[0012] According to an embodiment, performing the forward optimization comprises calculating a loss between (i) joint location prediction results and limb location prediction results generated by the neural network for a frame of video and (ii) a ground truth indication of joint locations and limb locations in the frame of video. Further, according to an embodiment, performing the backward optimization comprises processing, with the neural network, (i) joint location prediction results generated by the neural network for a frame of video, (ii) limb location prediction results generated by the neural network for the frame of video, and (iii) a previous frame to determine an indication of joint locations and an indication of limb locations for the previous frame. Such an embodiment calculates a loss between (i) the determined indication of joint locations and the determined indication of limb locations for the previous frame and (ii) a ground truth indication of joint locations and limb locations for the previous frame.

[0013] In yet another embodiment, performing forward and backward optimization between adjacent frames of video to refine joint location prediction results and limb location prediction results of the neural network comprises calculating a temporal consistency loss by calculating a loss between (i) joint location prediction results and limb location prediction results of the neural network for a first frame and (ii) joint location prediction results and limb location prediction results of the neural network for a second frame, wherein the second frame is adjacent to the first frame.

[0014] It is noted that embodiments of the method, system, and computer program product may be configured to implement any embodiments described herein.

BRIEF DESCRIPTION OF THE DRAWINGS

[0015] The foregoing will be apparent from the following more particular description of example embodiments, as illustrated in the accompanying drawings in which like reference characters refer to the same parts throughout the different views. The drawings are not necessarily to scale, emphasis instead being placed upon illustrating embodiments.

[0016] FIG. 1 is a simplified diagram of a system to identify joints and limbs according to an embodiment.

[0017] FIG. 2 is a flow diagram of a method for identifying joints and limbs in a frame of video according to an embodiment.

[0018] FIG. 3 is a block diagram of a framework for identifying joints and limbs in an embodiment.

[0019] FIG. 4 is a block diagram of a framework during a training phase according to an embodiment.

[0020] FIG. 5 is a block diagram of a system embodiment for identifying joints and limbs for a first frame of video.

[0021] FIG. 6 is a block diagram of a system embodiment for identifying joints and limbs using identification results from a previous frame.

[0022] FIG. 7 is a simplified block diagram of a computer system for identifying joints and limbs in a frame of video according to an embodiment.

[0023] FIG. 8 is a simplified diagram of a computer network environment in which an embodiment of the present invention may be implemented.

DETAILED DESCRIPTION

[0024] A description of example embodiments follows.

[0025] Pose estimation, which includes identifying joints and limbs in images and video, aims to estimate multiple poses of people or other such target objects in a frame of video and has been a long studied topic in computer vision [1, 6, 14, 9, 3] (bracketed numbers in this document refer to the enumerated list of references hereinbelow). Previous methods for human pose estimation utilized pictorial structures [1] or graphical models [3]. Recently, with the development and application of deep learning models, attempts to utilize deep convolutional neural networks to do 2D multi-person pose estimation have been made. These attempts can be categorized into two major categories, top-down methods and bottom-up methods.

[0026] Top-down methods detect persons by first using a person detector and then using single person pose estimation to get poses for all persons. He et al. [7] extended the Mask-RCNN framework to human pose estimation by predicting a one-hot mask for each body part. Papandreou et al. [11] utilized a Faster RCNN detector to predict person boxes and applied ResNet in a fully convolutional fashion to predict heatmaps for every body part. Fang et al. [5] designed a symmetric spatial transformer network to alleviate the inaccurate bounding box problem.

[0027] The existing top-down methods always utilize a separately trained person detector to first detect people in the image. With the knowledge of the detected people, i.e., bounding boxes of detected persons, top-down methods then do single-person keypoint estimation within each bounding box [7, 11, 5]. The problem with top-down methods is that if the person detection fails, the following keypoint estimation will also fail. Further, using two models, e.g., neural networks, in the top-down methods, also makes the top-down methods slower and makes utilizing top-down methods for real-time applications difficult.

[0028] Bottom-up methods do not utilize person detectors. Instead, bottom-up methods try to detect all of the body joints from the whole image and, then, associate those joints to each person to form their skeletons [12, 2, 10]. In general, bottom-up methods are less accurate compared to top-down methods. However, bottom-up methods can run faster than top-down methods in multi-person pose estimation. The inference time of bottom-up methods is less linear to the number of persons in the image.

[0029] Bottom-up methods detect body parts first and then associate body parts into persons. Insafutdinov et al. [12] proposed using an Inter Linear Program method to solve the body part association problem. Cao et al. [2] introduced Part Affinity Fields to predict the direction and activations for each limb to help associate body parts. Newell et al. [10] utilized predicted pixel-wise embeddings to assign detected body parts into different groups.

[0030] Video-based multi-person pose estimation often involves tracking methods as post processing. The post processing methods track the detected person across adjacent frames and then track the keypoints of that person to avoid detection failures caused by motion blur and occlusions. Those tracking methods cannot be applied to bottom-up methods because bottom-up methods do not provide any knowledge of a person in each frame. Tracking joints without knowing the movement of a person leads to unsatisfactory results. In video applications, bottom-up methods are applied on each frame, which leads to inconsistent pose estimation across adjacent frames. The inconsistency causes problems like shaking and jumping of keypoint detection.

[0031] Embodiments provide functionality for two-dimensional (2D) multi-person pose estimation in video. In embodiments, the pose estimation is formulated as detecting 2D keypoints, e.g., joints and limbs, and connecting the keypoints of the same person into skeletons. Embodiments provide a bottom-up method in multi-person pose estimation. Different from other methods, embodiments directly predict a confidence map for a human skeleton to associate the detected body parts.

[0032] Embodiments of the present invention provide a video-based state-of-the-art image-based bottom-up method for pose estimation that is specially optimized for video applications to solve the occluded and inconsistent detection between adjacent frames. To utilize the temporal information contained in the video and to avoid inconsistent detection across frames, embodiments use previous frames to refine the pose estimation result of the current frame. As such, embodiments track the poses across frames and use the determined results, e.g., pose, from a previous frame to refine the results for a current frame. By implementing this functionality, embodiments are resistant to pose occlusions. Moreover, embodiments build a backward path and reconstruct the previous pose estimation refined by the current estimation and minimize on the difference between the previous estimation and the reconstructed estimation. Assuming the movement between two adjacent frames is minor, an embodiment penalizes on the difference between the estimation on previous frame and the estimation on current frame to stabilize the pose estimation and alleviate any shaking and vibration on the predicted poses in the video.

[0033] Embodiments (1) utilize the pose estimation results of previous frames to refine the current frame results to track poses and handle occlusions (2) apply a backward loop to reconstruct the previous pose estimation from the current frames to minimize inconsistent detection and (3) penalize on the changes in detection between adjacent frames to avoid shaking and vibration in video pose estimation.

[0034] FIG. 1 illustrates a system 100 for identifying joints and limbs in a frame of video according to an embodiment. The system 100 includes the trained neural network 101. The neural network 101 is trained to identify joints and limbs in frame of video as described herein. In operation, the trained neural network 101 receives the frame 102 and processes the frame 102 to generate the indication 103 of joints, e.g., the joint 104, and limbs, e.g., the limb 105, in the frame 102. The trained neural network 101 may also generate the indication of joints and limbs 103 using an indication of joints and limbs that was determined for a frame prior in time to the frame 102.

[0035] FIG. 2 is a flow diagram of a method 220 for identifying joints and limbs in a current frame of video. The method 220, processes 221 the current frame of video to determine initial predictions of joint and limb locations in the current frame. In turn, the method 220, generates 222 indications of the joint and limb locations in the current frame by refining the initial predictions of the joint and limb locations based on indications of respective joint and limb locations from a previous frame. According to an embodiment, the indication of joint locations in the current frame indicates a probability of a joint at each location in the current frame and the indication of limb locations in the current frame indicates a probability of a limb at each location in the current frame. According to an embodiment, locations are x-y coordinates in the image. Further, in an embodiment, the unit of the locations, e.g., coordinates, are in pixels. Moreover, in an example embodiment, the previous frame is adjacent in time to the current frame in the video.

[0036] An embodiment of the method 220 further comprises generating an indication of pose for at least one object based upon the indications of the joint and limb locations in the current frame generated 222 for the current frame.

[0037] Embodiments of the method 220 may be used to identify limbs and joints of any type of object. For example, in an embodiment, the indications of the joint and limb locations in the current frame correspond to joints and limbs of at least one of: a human, animal, machine, and robot, amongst other examples. Moreover, embodiments may identify limbs and joints for multiple objects, e.g., people, in a frame.

[0038] According to an embodiment of the method 220, generating the indications of the joint and limb locations in the current frame 222 includes processing the initial prediction of joint locations in the current frame and the indications of joint locations from the previous frame with a first deep convolutional neural network to generate the indication of joint locations in the current frame. Further, in such an embodiment, the initial prediction of limb locations in the current frame and the indications of limb locations from the previous frame are processed with a second deep convolutional neural network to generate the indication of limb locations in the current frame.

[0039] Another embodiment of the method 220 identifies orientation of the identified limbs. Such an embodiment processes the current frame of video to determine an initial prediction of limb orientation at each initial prediction of limb location in the current frame. In turn, an indication of limb orientation in the current frame is generated by refining the initial prediction of limb orientation at each initial prediction of limb location in the current frame using indications of limb orientations from the previous frame. As such, the determination of limb orientation for a current frame is refined using the limb orientation results from a previous frame.

[0040] Hereinbelow, a problem formulation for limb and joint identification is provided and a framework for identifying joints and limbs according to an embodiment is described. Additional components of embodiments including joint prediction, limb prediction, backward reconstruction, temporal consistency, neural network training, and applying the trained neural network for video pose estimation are also further elaborated upon.

[0041] Problem Formulation

[0042] Let F_i be a frame sampled from a video sequence containing n frames $\{F_i\}_1^n$. Let $P_{i,j} = (\mathbf{x}_{i,j}, \mathbf{y}_{i,j})$ be the multi-person 2D pose keypoint coordinates of the jth person in the frame F_i. Given $\{F_i\}_1^k$ frames, where $0 < k \leq n$, which are the current frames and all the previous frames. In such an embodiment, the target is to estimate the current keypoints $\{\mathbf{P}_{k,j}\}_{j=1}^m$ where m is the number of persons in the current frame. Moreover, in the embodiment, a deep convolutional neural network model G takes the current frame and the previous frame as input and does pose estimation, which can be described as $\{\mathbf{P}_{k,j}\}_{j=1}^m = G(F_i, F_{i-1})$.

[0043] An implementedin of the neural network follows an image-based 2D bottom-up pose estimation method [2] to estimate a joint heatmap S_i and a limb heatmap L_i and, then, associates the joint and limb heatmaps into keypoint results $\mathbf{P}_{ij}$ using an association method denoted by M. Such an embodiment of the method can then be described by $\{\mathbf{P}_{kj}\}_{j=1}^m = M\big(G(F_i, S_{i-1}, L_{i-1})\big)$.

[0044] Framework

[0045] FIG. 3 illustrates a system framework 330 according to an embodiment. FIG. 3 illustrates the input, output, and the refinement process of the deep neural network model G 331. In FIG. 3, the variables 337 and 338 are from a previous frame, while the variables 332, 335, 336, 341, and 342 are for the current frame 332. F 332 is a frame sampled from a video. G_{S0} 333 and G_{L0} 334 are initial detection blocks, while G_{SR} 339 and G_{LR} 340 are refinement blocks for the joint heatmap S 335 and limb heatmap L 336.

[0046] In operation, G 331 takes the current frame 332 F_i as input and does an initial estimation using the submodules 333 G_{S0} and 334 G_{L0} to determine a joint heatmap 335 S_{i0} and limb heatmap 336 L_{i0}, respectively. In turn, the initial estimations 335 (joints) and 336 (limbs) are refined by the submodule 339 G_{SR} and submodule 340 G_{LR} using the previous results 337 S_{i-1} and 338 L_{i-1}. The refining by the submodules 339 and 340 produces 341 S_i and 342 L_i, which are the joint heatmap 341 and limb heatmap 342 for the frame 332 F_i. In the framework 330, 333 G_{S0}, 339 G_{SR}, 334 G_{L0}, and 340 G_{LR} are all deep convolutional neural networks. Further, in an embodiment, before inputing to 339 G_{SR} and 340 G_{LR}, 335 S_{i0}, 336 L_{i0}, 337 S_{i-1}, 338 L_{i-1} are concatenated together in channel dimension.

[0047] Joint Heatmap Prediction

[0048] In the joint heatmap prediction, the proposed framework, e.g., the framework 330, generates a confidence map, e.g., 341, which is the probability of joints appearing at each location of the input image, e.g., the frame 332. For an input image of size $H \times W \times 3$, the corresponding joint heatmap S will be of size $H \times W \times p$, where H and W are the height and width of the input image, and p is the number of joints.

[0049] To prepare a ground-truth heatmap prediction, e.g., the ground-truth predictions 441a and 442a discussed hereinbelow in relation to FIG. 4, an embodiment puts a gaussian response at each location of the same joints in the corresponding channel of the joint heatmap. The overlapping area of the same type of joints is handled by a maximum operation. The method to construct the ground-truth heatmap can be represented by equation (1) below:

$$\bar{\mathbf{S}}_i(x, y, c) = \max_{1 \leq j \leq m} (e^{-\frac{||(x,y)-\mathrm{P}_{i,j}^{l}||_2^2}{\sigma^2}}) \qquad (1)$$

where $P_{i,j}^{l}$ is the keypoints of the l-th joint of the j-th person in the i-th frame, and σ is the standard deviation of the Gaussion distribution.

[0050] An embodiment employs the idea of intermediate supervision such that the joint heatmap prediction output from G_{S0} and G_{SR} are compared with the ground-truth heatmap using a L2 loss function, which can be expressed as follows:

$$\mathcal{L}_S = ||G_{S0}(F_i) - \bar{S}_i||_2 + ||G_{SR}(G_{S0}(F_i), \bar{S}_{i-1}, \bar{L}_{i-1}) - \bar{S}_i||_2 \qquad (2)$$

[0051] In an embodiment, when minimizing the above joint prediction loss, the submodule G_{S0} and G_{SR} are trained to output the confidence map of the joint predictions for given images, i.e., frames.

[0052] Limb Prediction

[0053] For limb prediction, an embodiment predicts a vector field indicating the position and orientation of limbs in given frames. The prediction can also be seen as a confidence map with size $H \times W \times 2q$, where q is the number of limbs defined. To prepare the ground-truth confidence map for limb prediction, e.g., the ground truth predictions 441b and 442b discussed hereinbelow in relation to FIG. 4, an embodiment first defines q limbs between a pair of joints indicating meaningful human limbs (or limbs of any object being detected) such as head, neck, body, trunk and forearm, which will form a skeleton of a human body in the pose association part. Then, such an embodiment fills the region between those pairs of joints using a normalized vector pointing to the direction of those limbs. The limb region is defined as the points within distance of the line segment between a pair of joints. Numerically, such an embodiment defines the distance d from a point (x, y) to a limb segment of a pair of joints $((x_{l_1}, y_{l_1}), (x_{l_2}, y_{l_2}))$ as:

$$d = \frac{|(y_{l_1} - y_{l_2})x + (x_{l_2} - x_{l_1})y + (y_{l_1}x_{l_1} - y_{l_1}x_{l_2} - y_{l_2}x_{l_1} + y_{l_2}x_{l_2})|}{\sqrt{(y_{l_1} - y_{l_2})^2 + (x_{l_2} - x_{l_1})^2}} \quad (3)$$

[0054] The limb region comprises all the points in a rectangle where their distance d from the given limb is within a threshold θ, which represents half the width of the limb. Within the limb region, an embodiment fills each location in the limb region with the normalized vector of the limb denoted as:

$$\frac{\mathbf{p}_{l_1} - \mathbf{p}_{l_2}}{||\mathbf{p}_{l_1} - \mathbf{p}_{l_2}||_2} \quad (4)$$

[0055] Similar to joint prediction, an embodiment calculates a L2 loss between the predicted limb locations L_i and the ground-truth limb locations $\bar{L}_i$ as the objective function when training the framework. An embodiment sums up both the losses between intermediate prediction and refined prediction as the limb loss, which can be represented by:

$$\mathcal{L}_L = ||G_{L0}(F_i) - \bar{L}_i||_2 + ||G_{LR}(G_{L0}(F_i), S_{i-1}, L_{i-1}) - \bar{L}_i||_2 \quad (5)$$

[0056] <u>Backward Reconstruction</u>

[0057] Embodiments introduce a backward loop to reconstruct the joint heatmap and limb heatmap from the prediction in the current frame to increase the accuracy and robustness of inter-frame prediction. In detail, one such example embodiment inputs the current prediction and the previous frame to the neural network and predicts the joint heatmap and limb map of the previous frame. Then, such an embodiment compares the prediction with the ground-truth and calculates reconstruction losses of the joint heatmap and limb heatmap which can be expressed as follows:

$$\mathcal{L}_{Srec} = ||G_{SR}(G_{S0}(F_{i-1}), G_{SR}(G_{S0}(F_i), \bar{S}_{i-1}, \bar{L}_{i-1}), G_{LR}(G_{L0}(F_i), \bar{S}_{i-1}, \bar{L}_{i-1})) - \bar{L}_i||_2 \quad (6)$$

$$\mathcal{L}_{Lrec} = ||G_{LR}(G_{L0}(F_{i-1}), G_{SR}(G_{S0}(F_i), \bar{S}_{i-1}, \bar{L}_{i-1}), G_{LR}(G_{L0}(F_i), \bar{S}_{i-1}, \bar{L}_{i-1})) - \bar{L}_i||_2 \quad (7)$$

[0058] <u>Temporal Consistency</u>

[0059] To mitigate the shaking and vibration due to inconsistent detection between adjacent frames, an embodiment penalizes on the difference between two predictions generated for adjacent frames assuming that the frame rate is fast enough which indicates that the inter-frame movement is relatively small. Such an embodiment introduces temporal consistency loss which is the L2 loss between the predictions of adjacent frames, using the following equations:

$$\mathcal{L}_{Stemp} = ||G_{SR}(G_{S0}(F_i), \bar{S}_{i-1}, \bar{L}_{i-1}) - G_{SR}(G_{S0}(F_{i-1}), \bar{S}_i, \bar{L}_i)||_2 \qquad (8)$$

$$\mathcal{L}_{Ltemp} = ||G_{LR}(G_{L0}(F_i), \bar{S}_{i-1}, \bar{L}_{i-1}) - G_{LR}(G_{L0}(F_{i-1}), \bar{S}_i, \bar{L}_i)||_2 \qquad (9)$$

By minimizing the temporal consistency loss, such an embodiment minimizes the difference between two adjacent frames and obtains a stable prediction with minimum shaking and vibration.

[0060] FIG. 4 is a block diagram of a framework 440 during a training phase where a neural network of the framework 444 is trained to identify joints and limbs in a frame of video. The training in the framework 440 includes performing forward and backward optimization between adjacent frames of video to refine joint location prediction results and limb location prediction results of the neural network 444 and updating the neural network 444 based on the refined joint location prediction results and the refined limb location prediction results.

[0061] To perform the forward optimization, a current frame 443 and a ground truth indication of joint location 442a and a ground truth indication of limb location 442b, for a frame prior to the frame 443 (e.g., the frame 449) are processed by the neural network 444 to determine the indication of joint location 445 and limb location 446 for the frame 443. In turn, the loss 447 between (i) the joint location prediction results 445 and limb location prediction results 446 generated by the neural network 444 for the frame of video 443 and (ii) a ground truth indication of joint locations 441a and limb locations 441b in the frame of video 443 is calculated. Further, the loss 447 may be calculated with the binary mask 448 which masks out unlabeled regions in the frame 443. According to an embodiment, in the dataset (e.g., a dataset used to train the neural network 444), not every single person has a label. As such, embodiments may output joint and limb predictions of unlabeled persons in the video. However, those predictions do not have any ground-truth label to calculate losses. Thus, embodiments may use masks 448 and 453 to mask out those unlabeled persons. The masks 448 and 453 serve to disable those unlabeled areas when calculating the losses

According to an embodiment, the loss 447 is calculated as described hereinabove in relation to equations 2 and 5.

[0062] To perform the backward optimization in the framework 440, the neural network 444 processes (i) joint location prediction results 445 generated by the neural network 444 for the frame of video 443, (ii) limb location prediction results 446 generated by the neural network 444 for the frame of video 443, and (iii) a previous frame 449, to determine an indication of joint locations 450 and an indication of limb locations 451 for the previous frame 449. Then, the loss 452 is calculated. The loss 452 is the loss between (i) the determined indication of joint locations 450 and the determined indication of limb locations 451 for the previous frame 449 and (ii) a ground truth indication of joint locations 442a and limb locations 442b for the previous frame 449. Further, the loss 452 may be calculated with the binary mask 453 masking out unlabeled regions in the frame 449. According to an embodiment, the loss 452 is calculated as described hereinabove in relation to equations 6 and 7.

[0063] The framework 440 is also used to calculate the temporal consistency loss 454. The temporal consistency loss 454 is the loss between (i) the joint location prediction results 445 and limb location prediction results 446 of the neural network 444 for a first frame 443 and (ii) joint location prediction results 450 and limb location prediction results 451 of the neural network 444 for a second frame 449, wherein the second frame 449 is adjacent to the first frame 443. In an embodiment, the temporal consistency loss 454 is calculated as described hereinabove in relation to equations 8 and 9.

[0064] In an embodiment, the losses 447, 454, and 452 are used in the framework 440 to update and train the neural network 444. These losses 447, 454, and 452 may be implemented in Equation 10 as described hereinbelow. The losses 447, 454, and 452 are indications of errors in estimating the joint and limb locations. By minimizing the losses 447, 454, and 452, via the optimization process during training the neural network 444, the network 444 is trained to be more accurate on estimating the location of human body joints and limbs. The optimization process is done by mathematically updating the neural network 444 by descending the gradient of the overall objective. More detail of training the network can be found below.

[0065] Overall Objectives

[0066] In an embodiment, there is an overall objective function of current prediction loss, reconstruction loss, and temporal consistency loss, to optimize the proposed video 2D pose estimation neural network, which is denoted as

$$\mathcal{L} = \mathcal{L}_S + \mathcal{L}_L + \lambda_{rec}(\mathcal{L}_{Srec} + \mathcal{L}_{Lrec}) + \lambda_{temp}(\mathcal{L}_{Stemp} + \mathcal{L}_{Ltemp}) \quad (10)$$

where λ_{rec} and λ_{temp} are hyper-parameters which control the relative weights of the reconstruction loss and temporal consistency loss in the overall objective function. In an example implemtned, $\lambda_{rec} = 0.1$ and $\lambda_{temp} = 0.05$.

[0067] Below is a method for training the neural network for video 2D multi-person pose estimation with multi-frame refinement:

```
Initialize network parameters $\theta_G$
While $\theta_G$ has not converged do
    Sample a pair of adjacent frames and keypoints $\{(F_i, \mathbf{P}_i), (F_{i-1}, \mathbf{P}_{i-1})\}$ from data distribution $\mathbb{P}_{data}(F, \mathbf{P})$;
    Prepare ground-truth joints heatmaps $\bar{S}_i$ and $\bar{S}_{i-1}$ and limbs maps $\bar{L}_i$ and $\bar{L}_{i-1}$ using $\mathbf{P}_i$ and $\mathbf{P}_{i-1}$;
    Predict initial joints and limbs for both frames:
        $S_{0i}^{est} \leftarrow G_{S0}(F_i)$;
        $L_{0i}^{est} \leftarrow G_{L0}(F_i)$;
        $S_{0i-1}^{est} \leftarrow G_{S0}(F_{i-1})$;
        $L_{0i-1}^{est} \leftarrow G_{L0}(F_{i-1})$;
    Refine current frame results using previous frame ground-truth:
        $S_i^{est} \leftarrow G_{SR}(S_{0i}^{est}, L_{0i}^{est}, \bar{S}_{i-1}, \bar{L}_{i-1})$;
        $L_i^{est} \leftarrow G_{LR}(S_{0i}^{est}, L_{0i}^{est}, \bar{S}_{i-1}, \bar{L}_{i-1})$;
    Refine previous frame results using current frame ground-truth:
        $S_{i-1}^{est} \leftarrow G_{SR}(S_{0i-1}^{est}, L_{0i-1}^{est}, \bar{S}_i, \bar{L}_i)$;
        $L_{i-1}^{est} \leftarrow G_{LR}(S_{0i-1}^{est}, L_{0i-1}^{est}, \bar{S}_i, \bar{L}_i)$;
    Reconstruct previous frame results using current frame prediction:
        $S_{i-1}^{rec} \leftarrow G_{SR}(S_{0i-1}^{est}, L_{0i-1}^{est}, S_i^{est}, L_i^{est})$;
        $L_{i-1}^{rec} \leftarrow G_{LR}(S_{0i-1}^{est}, L_{0i-1}^{est}, S_i^{est}, L_i^{est})$;
    Calculate loss functions:
        $\mathcal{L}_S \leftarrow ||S_{0i}^{est} - \bar{S}_i||_2 + ||S_i^{est} - \bar{S}_i||_2$;
        $\mathcal{L}_L \leftarrow ||L_{0i}^{est} - \bar{L}_i||_2 + ||L_i^{est} - \bar{L}_i||_2$;
        $\mathcal{L}_{Srec} \leftarrow ||S_{i-1}^{rec} - \bar{S}_{i-1}||_2$;
        $\mathcal{L}_{Lrec} \leftarrow ||L_{i-1}^{rec} - \bar{L}_{i-1}||_2$;
        $\mathcal{L}_{Stemp} \leftarrow ||S_{i}^{est} - S_{i-1}^{est}||_2$;
        $\mathcal{L}_{Ltemp} \leftarrow ||L_{i}^{est} - L_{i-1}^{est}||_2$;
    Update $G$ by descending its gradient:
        $\nabla_{\theta_G}\mathcal{L}_S + \mathcal{L}_L + \lambda_{rec}(\mathcal{L}_{Srec} + \mathcal{L}_{Lrec}) + \lambda_{temp}(\mathcal{L}_{Stemp} + \mathcal{L}_{Ltemp})$;
End
Output: Converged model parameter $\theta_G$.
```

[0068] Training Method Embodiment

[0069] In the training phase, first, the data used for training the model is prepared. Then, a pair of adjacent frames with their ground-truth keypoints $(F_i, \mathbf{P}_i)$ and $(F_{i-1}, \mathbf{P}_{i-1})$ are randomly sampled from the data distribution $\mathcal{P}_{data}(F, \mathbf{P})$. F_i is of size $(H \times W \times 3)$, where H and W are the height of width of the frames. P_i is of size $m_i \times p \times 2$, where m_i is the number of people in the frame and p is the number of joints. For each type of joint, a Gaussian response is put in the joint heatmap S_i for each person in $\mathbf{P}_i$. In turn, S_i with size $H \times W \times p$ is obtained. The limbs are defined as the region between joints of a width within a threshold ε. For each limb region, such an embodiment fills each location with the limb direction denoted by a 2D normalized vector. Then, a limbs map of size $H \times W \times\times 2q$ is formed. S_i and L_i are downsampled to size $H/4 \times W/4 \times p$ and $H/4 \times W/4 \times 2q$ using nearest neighbor. After preparing the input frames and ground truth joints heatmap and ground truth limbs heatmap, the variables are fed to the framework and the overall objectives $\mathcal{L}$ are calculated. The network G is continuously updated by descending the gradient of the $\mathcal{L}$ using new pairs of data sampled from $\mathcal{P}_{data}(F, \mathbf{P})$.

[0070] Network Architecture

[0071] Table 1 below shows an example network architecture of a proposed pose estimation neural network that may be used in an embodiment.

[0072] The example deep convolutional neural network comprises a backbone, and four submodules as shown in Table 1. In Table 1, N=Number of filters, K=Kernel size, S=Stride, P=Padding, RELU=Rectified Linear Units, MAXPOOL2d=Max pooling operation for spatial data, p=Number of joints, and q=Number of limbs.

	Shared Backbone
1	CONV-(N64, K3, S1, P1, RELU)

2	CONV-(N64, K3, S1, P1, RELU)	
3	MAXPOOL2d(K2, S2, P0)	
4	CONV-(N128, K3, S1, P1, RELU)	
5	CONV-(N128, K3, S1, P1, RELU)	
6	MAXPOOL2d(K2, S2, P0)	
7	CONV-(N256, K3, S1, P1, RELU)	
8	CONV-(N256, K3, S1, P1, RELU)	
9	CONV-(N256, K3, S1, P1, RELU)	
10	CONV-(N128, K3, S1, P1, RELU)	
	G_{S0}	G_{L0}
11	CONV-(N128, K7, S1, P3, RELU)	CONV-(N128, K7, S1, P3, RELU)
12	CONV-(N128, K7, S1, P3, RELU)	CONV-(N128, K7, S1, P3, RELU)
13	CONV-(N512, K7, S1, P3, RELU)	CONV-(N512, K7, S1, P3, RELU)
14	CONV-(Np, K1, S1, P0, RELU)	CONV-(N2q, K1, S1, P0, RELU)
	G_{SR}	G_{LR}
15	CONV-(N128, K7, S1, P3, RELU)	CONV-(N128, K7, S1, P3, RELU)
16	CONV-(N128, K7, S1, P3, RELU)	CONV-(N128, K7, S1, P3, RELU)
17	CONV-(N128, K7, S1, P3, RELU)	CONV-(N128, K7, S1, P3, RELU)
18	CONV-(N128, K7, S1, P3, RELU)	CONV-(N128, K7, S1, P3, RELU)
19	CONV-(N512, K7, S1, P3, RELU)	CONV-(N512, K7, S1, P3, RELU)
20	CONV-(Np, K1, S1, P0, RELU)	CONV-(N2q, K1, S1, P0, RELU)

Table 1: Neural Network Architecture

[0073] The backbone is a VGG [13] style neural network used to extract pretrained features from a given frame. In an embodiment, the backbone is pretrained on ImageNet dataset [4] and fine-tuned in a pose estimation application. In the backbone, the input frame is downsampled twice with the MAXPOOL2d layer which reduces the height and width by 4 times when outputting the joints heatmap and limb heatmap. The backbone network is followed by a initial joint prediction submodule G_{S0} and a initial limb prediction module G_{L0}, which take the output of the backbone as their inputs and predict their results. After that, the prediction results are refined by the two refinement submodules G_{SR} and G_{LR}, which utilize multi-frame refinement to improve the accuracy and consistency of the prediction results. Embodiments provide a neural network that is lightweight and runs quickly on devices, such as GPU enabled devices. To further speed up operation, in an embodiment, the convolutional layers can be replaced by a pair of equivalent depthwise convolution layers and pointwise convolution layers such as the architecture proposed in MobileNet[8].

[0074] FIG. 5 is a block diagram of a system embodiment for identifying joints and limbs for a first frame of video. In the system 550, the identification of limbs and joints is made by the neural network 551 that includes the subnetworks 552, 553, 554, and 555, for a first frame of video 556.

[0075] In operation G 551 takes the current frame 556 as input and does an initial estimation using the submodules 552 and 553 to determine a joint heatmap 557 and a limb heatmap 558. In turn, the initial estimations 557 and 558 are refined by the submodule 554 and submodule 555 using the initial estimations themselves, 557 and 558. The refining by the submodules 554 and 555 produces 559 and 560, which are the estimation of the joint heatmap and limb heatmap of the frame 556. In this way, the system 550 implements self-refinement. In the framework 550, the submodules 552, 553, 554, and 555 are all deep convolutional neural networks.

[0076] The system 550 continues, and using the pose association module 561, constructs the one or more skeletons 562 in the frame 556 using both the joint prediction 559 and limb prediction 560. An embodiment may use pose association methods known in the art to assemble joints and limbs into skeletons.

[0077] FIG. 6 is a block diagram of a system embodiment 660 for identifying joints and limbs using identification results from a previous frame. In the system 660, the identification of limbs and joints is made by the neural network 661 that includes the subnetworks 662, 663, 664, and 665, for a frame of video 666.

[0078] In operation G 661 takes the current frame 666 as input and does an initial estimation using the submodule 662 and submodule 663 to determine a joint heatmap 667 and a limb heatmap 668. In turn, the initial estimations 667 and 668 are refined by the submodule 664 and submodule 665 using the joint estimation 673, i.e., heatmap, and limb estimation 674 from a previous frame of video. The refining by the submodules 664 and 665 produces the estimation of the joints heatmap 669 and the estimation of the limbs heatmap 670 for the frame 666. In this way, the system 660 refines the current estimation results 667 and 668 using the results 673 and 674 from a previous frame. In an embodiment, the refinement is done by the trained network 661 which includes the submodules 662, 663, 664, and 665. This refinement can handle difficult cases in video pose estimation such as motion blur and occulusion. The refinement can also improve the shaking and vibration of estimated results. In the framework 660, the submodules 662, 663, 664, and 665 are all deep convolutional neural networks.

[0079] The system 660 continues, and using the pose association module 671, constructs the one or more skeletons 672 in the frame 666 using both the joint prediction 669 and limb prediction 670.

[0080] Embodiments provide a novel deep learning model particularly optimized for video 2D multi-person pose estimation applications. Embodiments introduce multi-frame refinement and optimization to the bottom up pose estimation method. The multi-frame refinement and optimization includes a novel method of tracking, backward reconstruction, and temporal consistency. Multi-frame refinement enables the pose estimation model to track poses and handle occlusions. Backward reconstruction and temporal consistency minimize inconsistent detection, which mitigates the shaking and vibration and improves the robustness in video pose estimation applications.

[0081] Using multi-frame refinement as described herein can be considered as an equivalent process to tracking. Tracking is a method to refine results by considering the temporal movement of objects in the video. Traditionally, approaches use the final output results of pose estimation to do tracking based on statistic assumptions. Tracking methods often stabilize the estimation results and improve the accuracy. Embodiments train the neural network to learn the movement of human bodies by feeding the neural network with previous frames. Then, the neural network can track the poses from previous frames and estimate the current poses more accurately even under occlusions. Embodiments can also enforce temporal consistency between adjacent frames to stabilize the results. As such, embodiments can provide tracking by multi-frame refinement.

[0082] Embodiments tackle a video-based multi-person pose estimation problem using a deep learning framework with multi-frame refinement and optimization. In a particular embodiment, a method inherently tracks estimated poses and makes a model insensitive to occlusions. The method may employ a backward reconstruction loop and temporal consistency to an objective function that mitigates inter-frame inconsistency and significantly reduces shaking and vibration phenomena of estimated pose skeletons in video pose estimation.

[0083] An embodiment of the invention utilizes pose estimation results of previous frames to refine a current frame result to track poses and handle occlusions. An embodiment of the invention applies a backward loop to reconstruct a previous pose estimation from a current frame to improve robustness and minimize inconsistent estimation. An embodiment of the invention introduces a temporal consistency loss that penalizes on temporal changes in detection between adjacent frames to avoid shaking and vibration in video pose estimation.

[0084] Embodiments generate a more accurate and robust pose estimation than existing methods. An embodiment tracks multi-person human poses in videos and handles occlusions. Embodiments output pose estimation with temporal consistency across frames, which avoids shaking and vibration in video pose estimation. Embodiments are computationally less expensive compared to the other pose estimation methods which require extra tracking modules.

[0085] Embodiments can be applied in detecting human behaviors in monitoring systems. Embodiments can be applied in video games to use human body movement as input, such as Xbox® Kinect®. Embodiments can be applied in many interesting mobile apps that require human body movement as input such as personal fitting and training.

[0086] Video-based multi-person pose estimation often involves tracking methods to improve estimation accuracy by utilizing temporal information in videos. The tracking methods track a detected person across adjacent frames and then track key points of that person to avoid failure detection due to motion blur and occlusions. Those tracking methods cannot be applied on bottom-up methods since bottom-up methods do not provide any knowledge of the person in each frame. Tracking the person's joints (e.g., elbows shoulders, knees) without knowing the movement of the person leads to unsatisfactory results. In video applications, pose estimation is applied frame by frame, which leads to inconsistent pose estimation across adjacent frames. The inconsistency causes problems, like shaking and jumping of key point detection.

[0087] To solve the above problems, an embodiment of the invention of video multi-person pose estimation provides a state-of-the-art image-based bottom-up method that is specially optimized for a video application to solve the inconsistent detection between adjacent frames. To utilize the temporal information contained in the video and avoid inconsistent detection across frames, a previous frame is used to refine a pose estimation result of a current frame. An embodiment tracks the person's poses across frames to handle occlusions. Another embodiment builds a backward path and reconstructs a previous pose estimation refined by a current estimation and penalizes on inconsistency between adjacent pose estimation. Moreover, assuming the movement between two adjacent frames is minor, an embodiment also penalizes based on a difference between the estimation on a previous frame and the estimation on a current frame to stabilize the pose estimation and alleviate shaking and vibration of the estimated poses in videos. With the above techniques, embodiments establish a robust and stable multi-person pose estimation which can be deployed on many applications that require human pose input.

[0088] In an embodiment, the input joints location are results from the previous frame. The neural network takes the estimation from the previous frame to help estimate the joints location of the current frame. The refined results here are referred to results of the current frame. By comparing the results with the ground-truth location an embodiment can update the network to correctly predict the joints locations of current frames.

[0089] FIG. 7 is a simplified block diagram of a computer-based system 770 that may be used to implement any variety of the embodiments of the present invention described herein. The system 770 comprises a bus 773. The bus 773 serves as an interconnect between the various components of the system 770. Connected to the bus 773 is an input/output device interface 776 for connecting various input and output devices such as a keyboard, mouse, display, speakers, etc. to the system 770. A central processing unit (CPU) 772 is connected to the bus 773 and provides for the execution of computer instructions implementing embodiments. Memory 775 provides volatile storage for data used for carrying out computer instructions implementing embodiments described herein, such as those embodiments previously described hereinabove. Storage 774 provides non-volatile storage for software instructions, such as an operating system (not shown) and embodiment configurations, etc. The system 770 also comprises a network interface 771 for connecting to any variety of networks known in the art, including wide area networks (WANs) and local area networks (LANs).

[0090] It should be understood that the example embodiments described herein may be implemented in many different ways. In some instances, the various methods and systems described herein may each be implemented by a physical, virtual, or hybrid general purpose computer, such as the computer system 770, or a computer network environment such as the computer environment 880, described herein below in relation to FIG. 8. The computer system 770 may be transformed into the systems that execute the methods described herein, for example, by loading software instructions into either memory 775 or non-volatile storage 774 for execution by the CPU 772. One of ordinary skill in the art should further understand that the system 770 and its various components may be configured to carry out any embodiments or combination of embodiments of the present invention described herein.

Further, the system 770 may implement the various embodiments described herein utilizing any combination of hardware, software, and firmware modules operatively coupled, internally, or externally, to the system 770.

[0091] FIG. 8 illustrates a computer network environment 880 in which an embodiment of the present invention may be implemented. In the computer network environment 880, the server 881 is linked through the communications network 882 to the clients 883a-n. The environment 880 may be used to allow the clients 883a-n, alone or in combination with the server 881, to execute any of the embodiments described herein. For non-limiting example, computer network environment 880 provides cloud computing embodiments, software as a service (SAAS) embodiments, and the like.

[0092] Embodiments or aspects thereof may be implemented in the form of hardware, firmware, or software. If implemented in software, the software may be stored on any non-transient computer readable medium that is configured to enable a processor to load the software or subsets of instructions thereof. The processor then executes the instructions and is configured to operate or cause an apparatus to operate in a manner as described herein.

[0093] Further, firmware, software, routines, or instructions may be described herein as performing certain actions and/or functions of the data processors. However, it should be appreciated that such descriptions contained herein are merely for convenience and that such actions in fact result from computing devices, processors, controllers, or other devices executing the firmware, software, routines, instructions, etc.

[0094] It should be understood that the flow diagrams, block diagrams, and network diagrams may include more or fewer elements, be arranged differently, or be represented differently. But it further should be understood that certain implementations may dictate the block and network diagrams and the number of block and network diagrams illustrating the execution of the embodiments be implemented in a particular way.

[0095] Accordingly, further embodiments may also be implemented in a variety of computer architectures, physical, virtual, cloud computers, and/or some combination thereof, and thus, the data processors described herein are intended for purposes of illustration only and not as a limitation of the embodiments.

[0096] The teachings of all patents, published applications and references cited herein are incorporated by reference in their entirety.

[0097] While example embodiments have been particularly shown and described, it will be understood by those skilled in the art that various changes in form and details may be made therein without departing from the scope of the embodiments encompassed by the appended claims.

[0098] References

[0099] [1] M. Andriluka, S. Roth, and B. Schiele. Pictorial structures revisited: People detection and articulated pose estimation. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1014–1021. IEEE, 2009.

[00100] [2] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7291–7299, 2017.

[00101] [3] X. Chen and A. L. Yuille. Articulated pose estimation by a graphical model with image dependent pairwise relations. In *Advances in neural information processing systems*, pages 1736–1744, 2014.

[00102] [4] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

[00103] [5] H. Fang, S. Xie, Y.-W. Tai, and C. Lu. Rmpe: Regional multi-person pose estimation. In *The IEEE International Conference on Computer Vision (ICCV)*, volume 2, 2017.

[00104] [6] G. Gkioxari, P. Arbeláez, L. Bourdev, and J. Malik. Articulated pose estimation using discriminative armlet classifiers. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 3342–3349. IEEE, 2013.

[00105] [7] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, pages 2980–2988. IEEE, 2017.

[00106] [8] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

[00107] [9] S. Johnson and M. Everingham. Learning effective human pose estimation from inaccurate annotation. In *Computer vision and pattern recognition (CVPR), 2011 IEEE conference on*, pages 1465–1472. IEEE, 2011.

[00108] [10] A. Newell, Z. Huang, and J. Deng. Associative embedding: End-to-end learning for joint detection and grouping. In *Advances in Neural Information Processing Systems*, pages 2274–2284, 2017.

[00109] [11] G. Papandreou, T. Zhu, N. Kanazawa, A. Toshev, J. Tompson, C. Bregler, and K. Murphy. Towards accurate multi-person pose estimation in the wild. In *Proceedings* *of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4903–4911, 2017.

[00110] [12] L. Pishchulin, E. Insafutdinov, S. Tang, B. Andres, M. Andriluka, P. V. Gehler, and B. Schiele. Deepcut: Joint subset partition and labeling for multi person pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4929–4937, 2016.

[00111] [13] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[00112] [14] Y. Yang and D. Ramanan. Articulated pose estimation with flexible mixtures-of-parts. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1385–1392. IEEE, 2011.

CLAIMS

What is claimed is:

1. A method of identifying joints and limbs in a current frame of video, the method comprising:

processing the current frame of video to determine initial predictions of joint and limb locations in the current frame; and

generating indications of the joint and limb locations in the current frame by refining the initial predictions of the joint and limb locations based on indications of respective joint and limb locations from a previous frame.

2. The method of Claim 1 further comprising:

generating an indication of pose for at least one object based upon the indications of the joint and limb locations in the current frame.

3. The method of Claim 1 wherein the indications of the joint and limb locations in the current frame correspond to joints and limbs of at least one of: a human, animal, machine, and robot.

4. The method of Claim 1 wherein generating the indications of the joint and limb locations in the current frame comprises:

processing the initial prediction of joint locations in the current frame and the indications of joint locations from the previous frame with a first deep convolutional neural network to generate the indication of joint locations in the current frame; and

processing the initial prediction of limb locations in the current frame and the indications of limb locations from the previous frame with a second deep convolutional neural network to generate the indication of limb locations in the current frame.

5. The method of Claim 1 wherein:

the indication of joint locations in the current frame indicates a probability of a joint at each location in the current frame; and the indication of limb locations in the current frame indicates a probability of a limb at each location in the current frame.

6. The method of Claim 1 further comprising:

processing the current frame of video to determine an initial prediction of limb orientation at each initial prediction of limb location in the current frame; and

generating an indication of limb orientation in the current frame by refining the initial prediction of limb orientation at each initial prediction of limb location in the current frame using indications of limb orientations from the previous frame.

7. The method of Claim 1 wherein the previous frame is adjacent in time to the current frame in the video.

8. A computer system for identifying joints and limbs in a current frame of video, the computer system comprising:

a processor; and

a memory with computer code instructions stored thereon, the processor and the memory, with the computer code instructions, being configured to cause the system to:

process the current frame of video to determine initial predictions of joint and limb locations in the current frame; and

generate indications of the joint and limb locations in the current frame by refining the initial predictions of the joint and limb locations based on indications of respective joint and limb locations from a previous frame.

9. The system of Claim 8 wherein the processor and the memory, with the computer code instructions, are further configured to cause the system to:

generate an indication of pose for at least one object based upon the indications of the joint and limb locations in the current frame.

10. The system of Claim 8 wherein the indications of joint and limb locations in the current frame correspond to joints and limbs of at least one of:

a human, animal, machine, and robot.

11. The system of Claim 8 wherein, in generating the indications of the joint and limb locations in the current frame, the processor and the memory, with the computer code instructions, are configured to cause the system to:

process the initial prediction of joint locations in the current frame and the indications of joint locations from the previous frame with a first deep convolutional neural network to generate the indication of joint locations in the current frame; and

process the initial prediction of limb locations in the current frame and the indications of limb locations from the previous frame with a second deep convolutional neural network to generate the indication of limb locations in the current frame.

12. The system of Claim 8 wherein:

the indication of joint locations in the current frame indicates a probability of a joint at each location in the current frame; and

the indication of limb locations in the current frame indicates a probability of a limb at each location in the current frame.

13. The system of Claim 8 wherein the processor and the memory, with the computer code instructions, are further configured to cause the system to:

process the current frame of video to determine an initial prediction of limb orientation at each initial prediction of limb location in the current frame; and

generate an indication of limb orientation in the current frame by refining the initial prediction of limb orientation at each initial prediction of limb location in the current frame using indications of limb orientations from the previous frame.

14. The system of Claim 8 wherein the previous frame is adjacent in time to the current frame in the video.

15. A computer program product for identifying joints and limbs in a current frame of video, the computer program product comprising:

one or more non-transitory computer-readable storage devices and program instructions stored on at least one of the one or more storage devices, the program instructions, when loaded and executed by a processor, cause an apparatus associated with the processor to:

process the current frame of video to determine initial predictions of joint and limb locations in the current frame; and

generate indications of the joint and limb locations in the current frame by refining the initial predictions of the joint and limb locations based on indications of respective joint and limb locations from a previous frame.

16. The computer program product of Claim 15 wherein the program instructions, when loaded and executed by the processor, further cause the apparatus associated with the processor to:

process the current frame of video to determine an initial prediction of limb orientation at each initial prediction of limb location in the current frame; and

generate an indication of limb orientation in the current frame by refining the initial prediction of limb orientation at each initial prediction of limb location in the current frame using indications of limb orientations from the previous frame.

17. A method of training a neural network to identify joints and limbs in a current frame of video, the method comprising:

performing forward and backward optimization between adjacent frames of video to refine joint location prediction results and limb location prediction results of a neural network; and

updating the neural network based on the refined joint location prediction results and the refined limb location prediction results.

18. The method of Claim 17 wherein performing the forward optimization comprises:

calculating a loss between (i) joint location prediction results and limb location prediction results generated by the neural network for a frame of video and (ii) a ground truth indication of joint locations and limb locations in the frame of video.

19. The method of Claim 17 wherein performing the backward optimization comprises:

processing, with the neural network, (i) joint location prediction results generated by the neural network for a frame of video, (ii) limb location prediction results generated by the neural network for the frame of video, and (iii) a previous frame to determine an indication of joint locations and an indication of limb locations for the previous frame; and

calculating a loss between (i) the determined indication of joint locations and the determined indication of limb locations for the previous frame and (ii) a ground truth indication of joint locations and limb locations for the previous frame.

20. The method of Claim 17 wherein performing forward and backward optimization between adjacent frames of video to refine joint location prediction results and limb location prediction results of the neural network comprises:

calculating a temporal consistency loss by calculating a loss between (i) joint location prediction results and limb location prediction results of the neural network for a first frame and (ii) joint location prediction results and limb location prediction results of the neural network for a second frame, wherein the second frame is adjacent to the first frame.

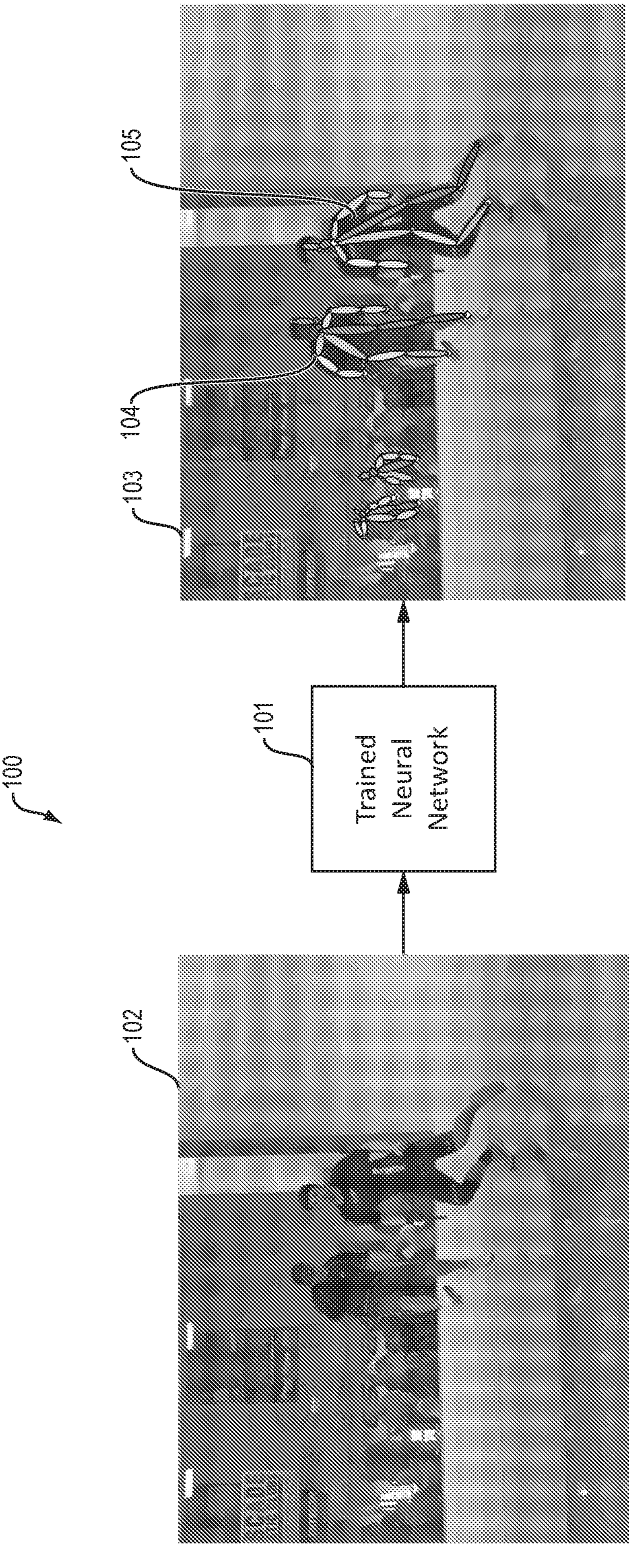

FIG. 1

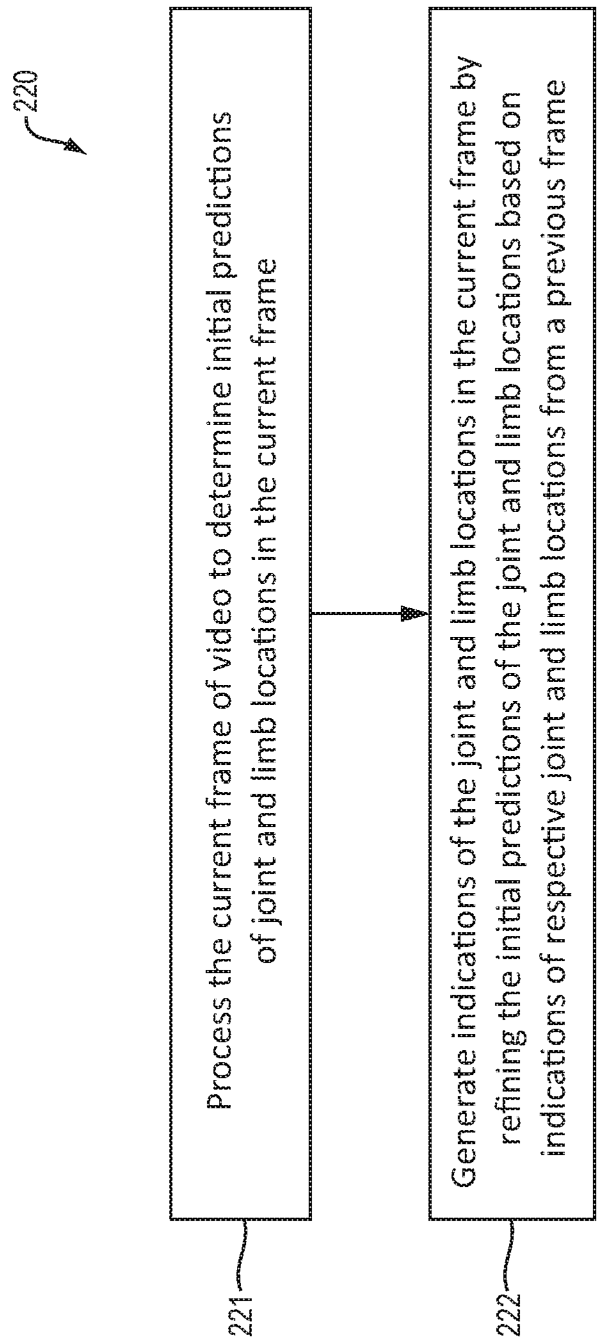
220
Process the current frame of video to determine initial predictions of joint and limb locations in the current frame
221
Generate indications of the joint and limb locations in the current frame by refining the initial predictions of the joint and limb locations based on indications of respective joint and limb locations from a previous frame
222

FIG. 2

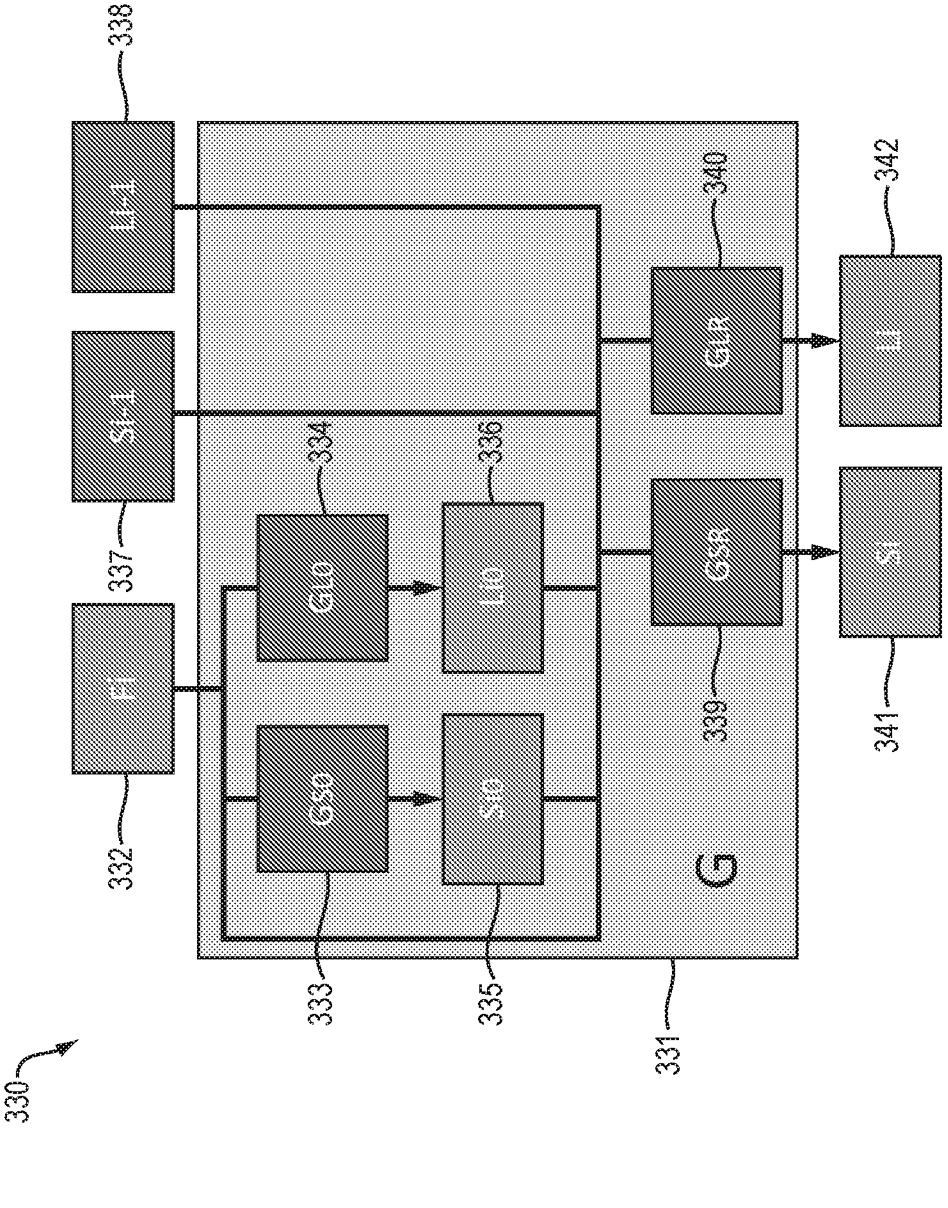
330
331
332
333
334
335
336
337
338
339
340
341
342
Ft
St-1
Lt-1
GS0
GL0
St0
Lt0
GSR
GLR
St
Lt
G

FIG. 3

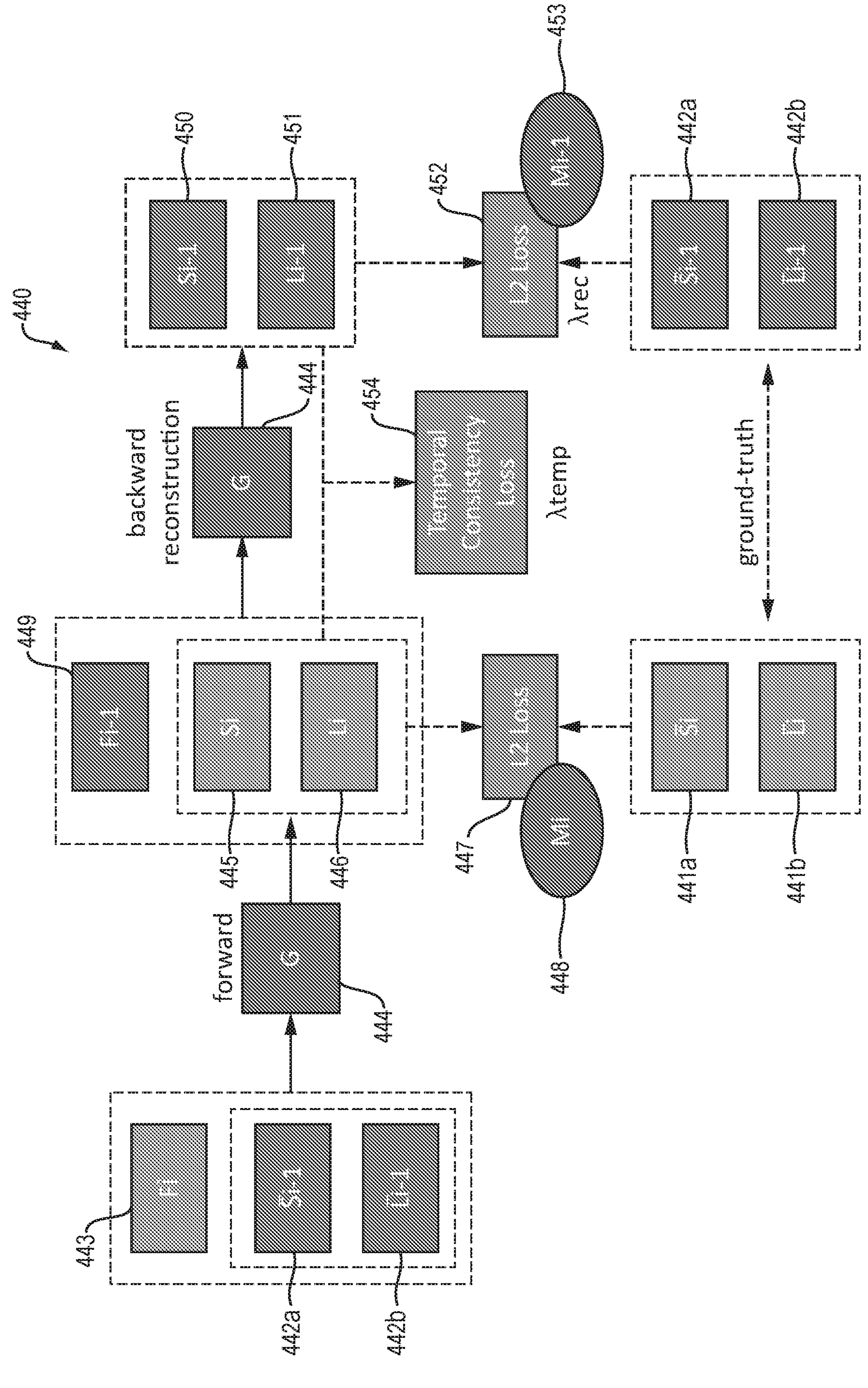
440
443
Fi
Si-1
Li-1
442a
442b
forward
G
444
445
446
449
Fi-1
Si
Li
backward reconstruction
G
444
450
451
Si-1
Li-1
454
Temporal Consistency loss
λtemp
452
L2 Loss
Mi-1
453
λrec
447
L2 Loss
Mi
448
Ŝi
L̂i
441a
441b
ground-truth
Ŝi-1
L̂i-1
442a
442b

FIG. 4

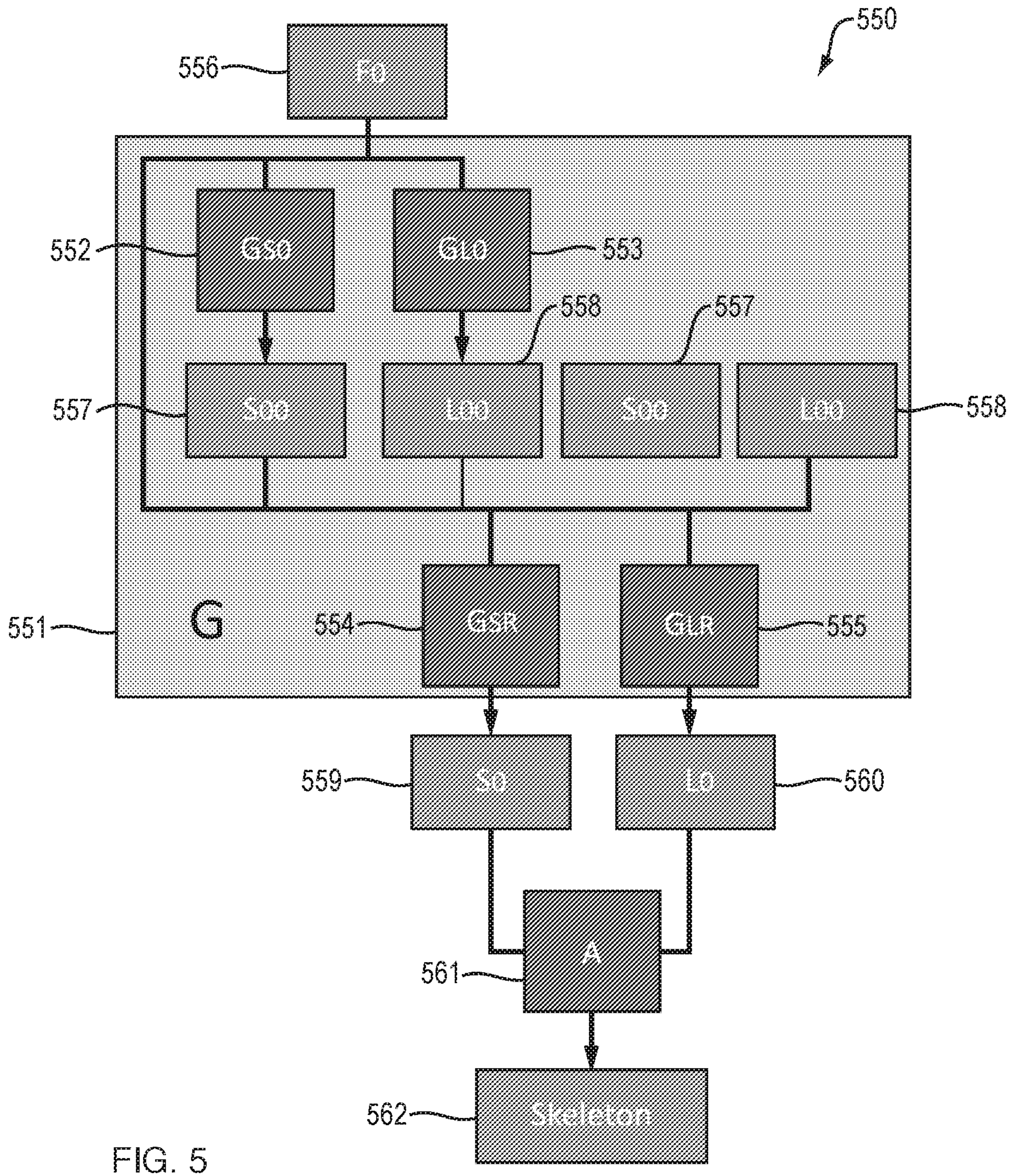
550
556
F0
552
GS0
GL0
553
558
557
557
S00
L00
S00
L00
558
551
G
554
GSR
GLR
555
559
S0
L0
560
561
A
562
Skeleton

FIG. 5

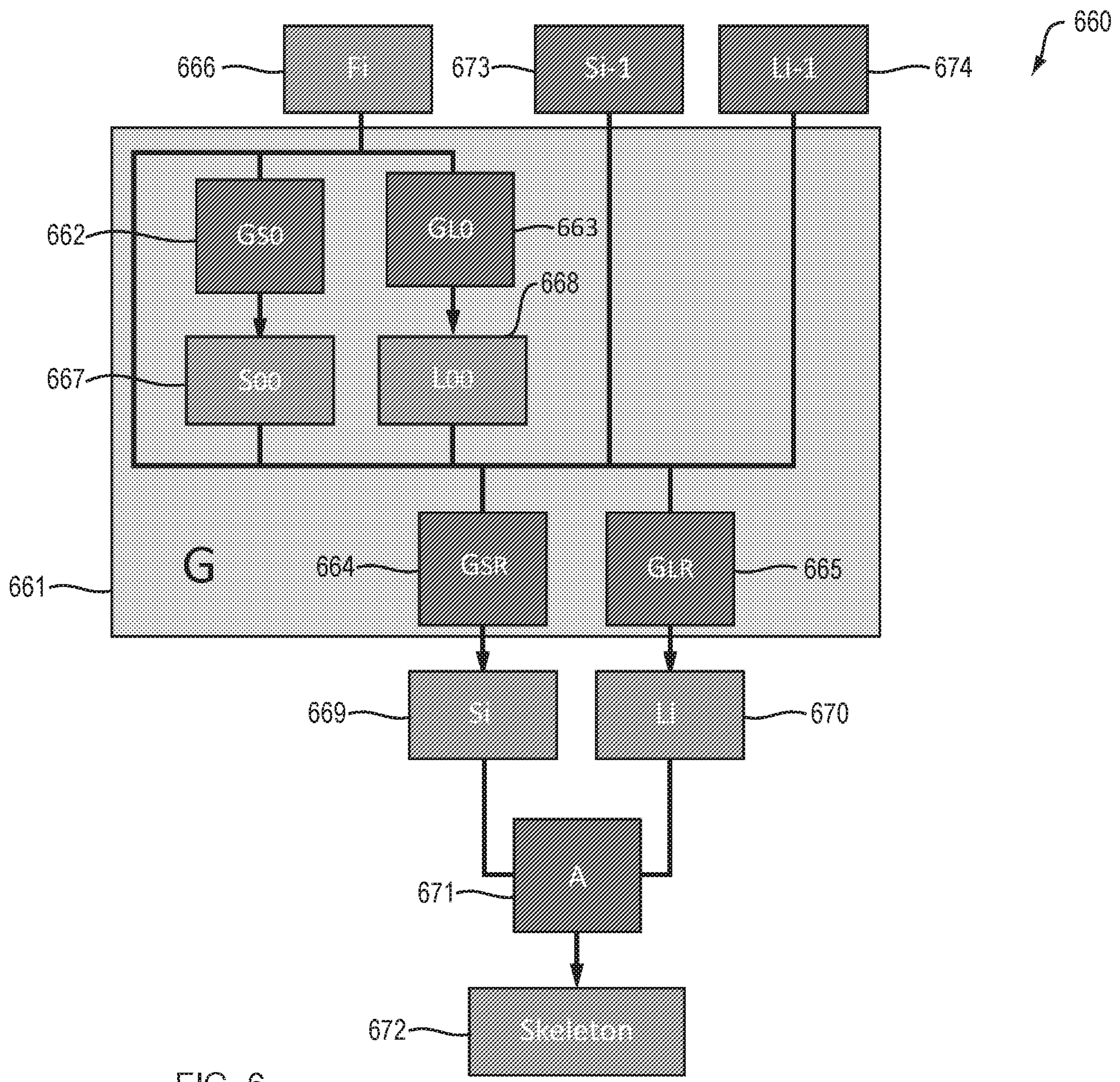
660
Fi
666
Si-1
673
Li-1
674
GS0
662
GL0
663
668
S00
667
L00
G
661
GSR
664
GLR
665
Si
669
Li
670
A
671
Skeleton
672

FIG. 6

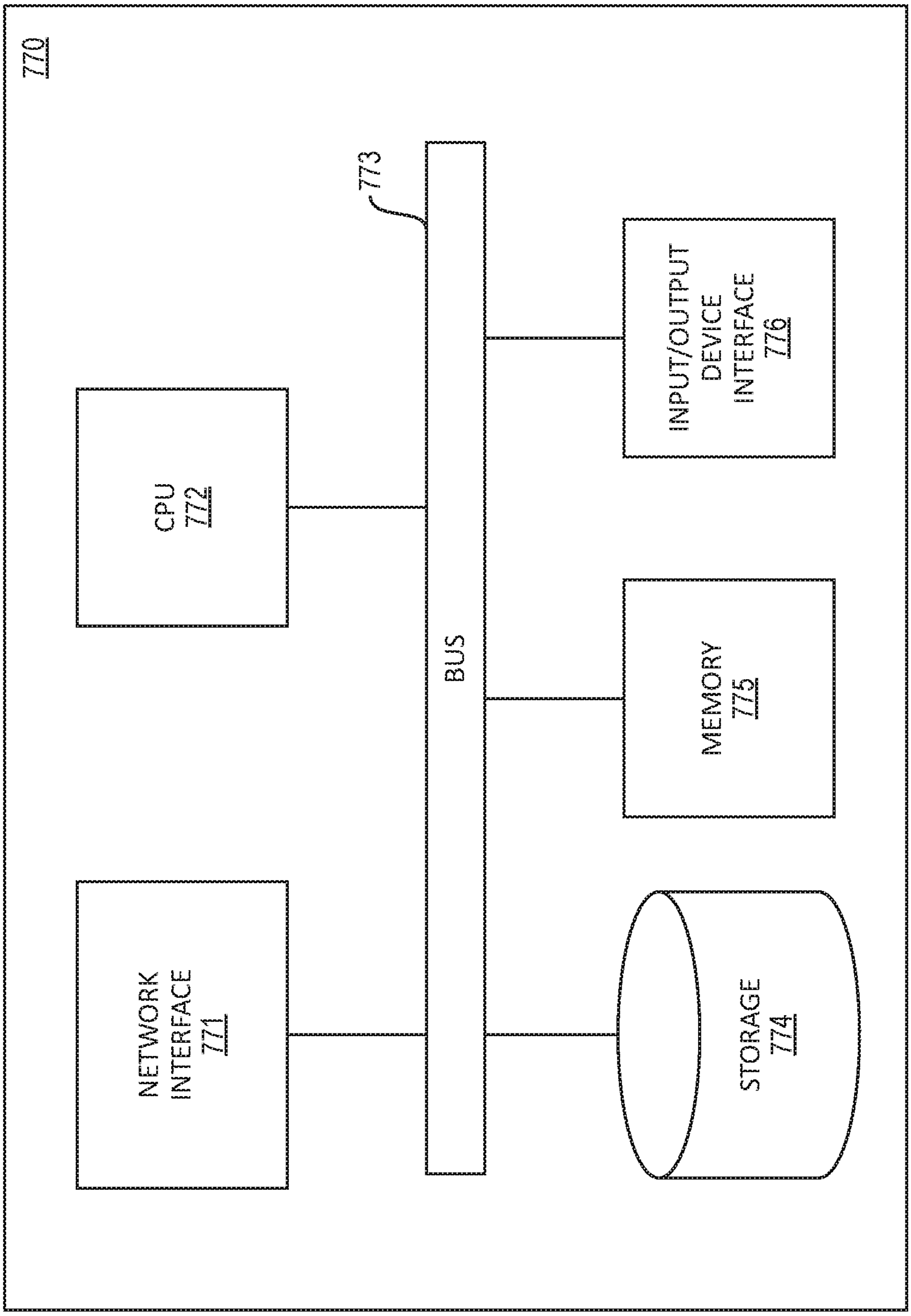
770
NETWORK INTERFACE 771
CPU 772
BUS
773
STORAGE 774
MEMORY 775
INPUT/OUTPUT DEVICE INTERFACE 776

FIG. 7

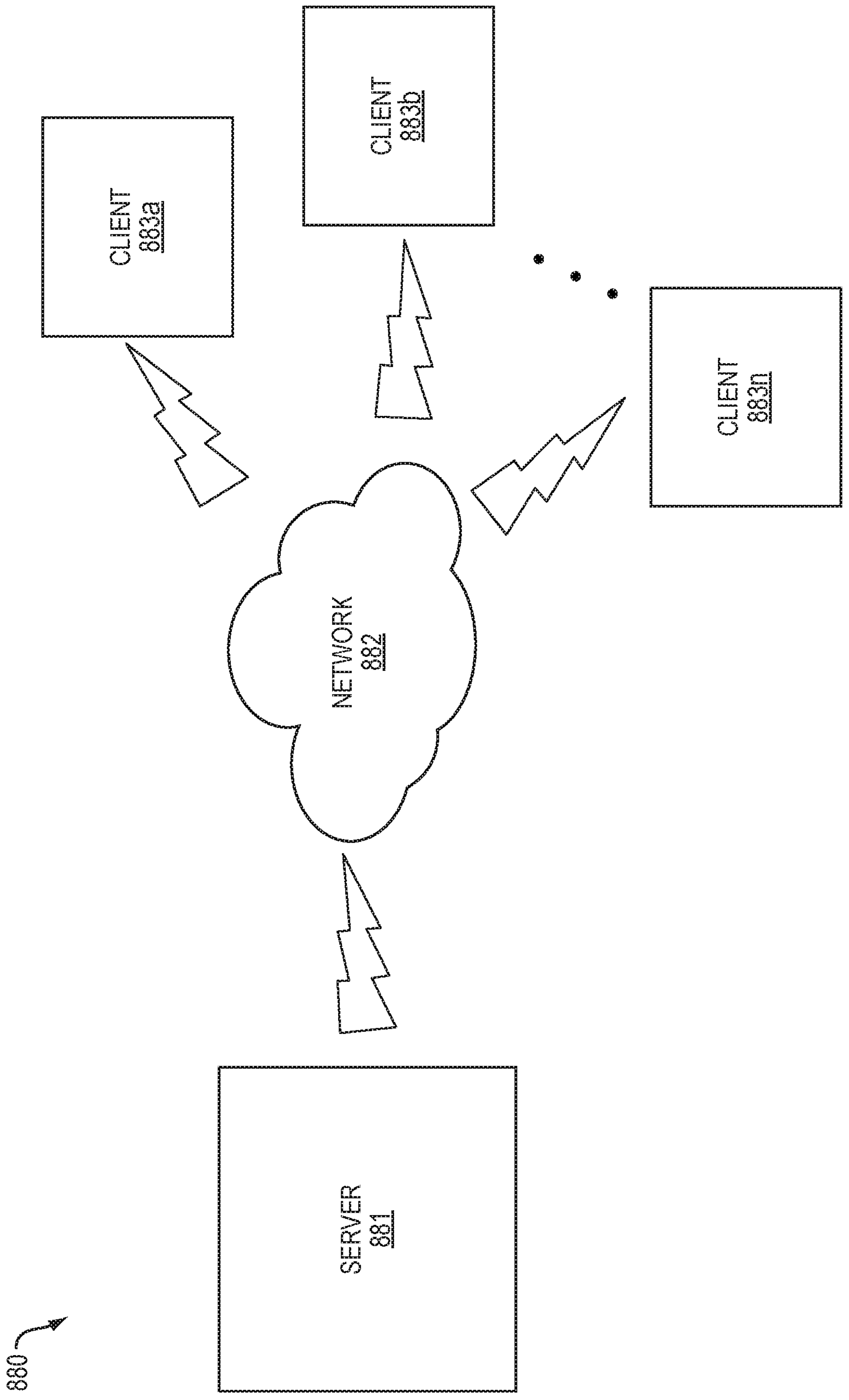
CLIENT
883a
CLIENT
883b
CLIENT
883n
NETWORK
882
SERVER
881
880

FIG. 8

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2020/032595

A. CLASSIFICATION OF SUBJECT MATTER
INV. G06T7/70
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)
G06T G06K

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

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	JIE SONG ET AL: "Thin-Slicing Network: A Deep Structured Model for Pose Estimation in Videos", ARXIV.ORG, CORNELL UNIVERSITY LIBRARY, 201 OLIN LIBRARY CORNELL UNIVERSITY ITHACA, NY 14853, 31 March 2017 (2017-03-31), XP080957029,	1-5, 7-12,14, 15
Y	abstract; figure 2	17
A	page 2, left-hand column, line 4 - line 31 -----	18-20
	-/--	

☒ 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 29 June 2020	Date of mailing of the international search report 07/07/2020
Name and mailing address of the ISA/ European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Authorized officer Rimassa, Simone

1

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	ELDAR INSAFUTDINOV ET AL: "Articulated Multi-person Tracking in the Wild", ARXIV.ORG, CORNELL UNIVERSITY LIBRARY, 201 OLIN LIBRARY CORNELL UNIVERSITY ITHACA, NY 14853, 5 December 2016 (2016-12-05), XP080736916, DOI: 10.1109/CVPR.2017.142	1-5, 7-12,14, 15
Y	abstract; figures 4,5	17
A	section Temporal Model -----	18-20
X	CHARLES JAMES ET AL: "Personalizing Human Video Pose Estimation", 2016 IEEE CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION (CVPR), IEEE, 27 June 2016 (2016-06-27), pages 3063-3072, XP033021488, DOI: 10.1109/CVPR.2016.334 [retrieved on 2016-12-09]	1-5, 7-12,14, 15
Y	figure 6	17
A	section Annotation propagation and refinement -----	18-20
X	MIR RAYAT IMTIAZ HOSSAIN ET AL: "Exploiting temporal information for 3D pose estimation", ARXIV.ORG, CORNELL UNIVERSITY LIBRARY, 201 OLIN LIBRARY CORNELL UNIVERSITY ITHACA, NY 14853, 23 November 2017 (2017-11-23), XP081073271, DOI: 10.1007/978-3-030-01249-6_5	1-16,20
Y	abstract	17
A	second paragraph of page 3; section Loss Function -----	18,19
Y	YI YANG ET AL: "Single online visual object tracking with enhanced tracking and detection learning", MULTIMEDIA TOOLS AND APPLICATIONS, KLUWER ACADEMIC PUBLISHERS, BOSTON, US, vol. 78, no. 9, 23 October 2018 (2018-10-23), pages 12333-12351, XP036779958, ISSN: 1380-7501, DOI: 10.1007/S11042-018-6787-6 [retrieved on 2018-10-23] abstract bottom of page 12334 item (1) ----- -/--	17

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	Christian Payer ET AL: "Simultaneous Multi-Person Detection and Single-Person Pose Estimation With a Single Heatmap Regression Network", , 29 October 2017 (2017-10-29), XP055642445, Retrieved from the Internet: URL:https://pdfs.semanticscholar.org/aa65/78f40975ecdc0d80af6941bd22403f06abff.pdf [retrieved on 2019-11-14] abstract -----	1,8,15

1