Multiple Human's Pose Detection Algorithm for Real Time Video

Aliaksandr Leunikau
Belarusian State University
Minsk, Belarus
Alex.levnikov@gmail.com

Alexander Nedzved
FAMCS of
Belarusian State University
Minsk, Belarus
NedzvedA@tut.by
ORCID 0000-0001-6367-5900

Stanislav Sholtanyuk
Belarusian State University
Minsk, Republic of Belarus
ssholtanyuk@bsu.by
ORCID 0000-0003-0266-7135

Alexei Belotserkovsky
Dept. of Intelligent Information Systems
United Institute of Informatics Problems of NAS of Belarus
Minsk, Belarus
alex.belot@gmail.com

Abstract. **This article describes a realtime algorithm to determine a person's posture at a certain point in time.**

Keywords: **human's pose recognition, virtual human skeleton, object movement analysis, neural networks, image recognition**

I. Introduction

The pace of development of technology and robotics in the modern world is striding far into the future, but there are still many unsolved problems assigned exclusively to humans. Isolation of familiar faces from the environment, determination of the character and state of a person by non-verbal gestures and facial expressions, determination of emotions, type of activity, and occupation. All of these tasks are considered purely human. But thanks to the non-linear advances in technology, computer vision, machine learning, and artificial intelligence, robotics and almost any electronic device equipped with sufficient resources can learn how to solve these problems.

This research is aimed at solving creative, inherent only to humans, problems. Specifically, tasks related to pattern recognition and positioning of objects in space.

This topic is relevant for many areas of human life, from mass media systems to medicine and security. The algorithm can be used to analyze gestures in sign language translation, analyze people's behavior by security cameras, overlay animations in the film and game industries.

The purpose of this work is to develop a new algorithm for analyzing the positioning of people in space, capable of increasing the efficiency of solving problems of image recognition in images.

II. Problem Review

Assessing a person's positioning is a problem of localizing anatomical key points, later called body parts, which in physical terms are the joints points of the human body, with the exception of the face key points. This problem is mainly focused on finding body parts, combining them into a complete skeleton, and tracking it throughout the video sequence.

Recognizing the poses of several people in an image, especially socially active ones, presents a unique set of challenges:

- Each image can contain an unknown number of people appearing and disappearing at any position and scale.
- Interactions between people cause complex spatial interference, which is caused by occlusions, physical contacts between people, properties of joints, which greatly complicates the unification of individual parts of the body in the limb, and later into the skeleton.
- The complexity of execution increases with the number of people, which imposes significant restrictions on the mode of execution in real-time.

III. Solution Review

The algorithm is a bottom-up method for estimating associations through part affinity fields, a set of two-dimensional vector fields that encode the location and orientation of limbs in an image region.

The algorithm has three main steps:

- Image preprocessing.
- Simultaneous body parts detection and association.

Pattern Recognition and Information Processing (PRIP'2021) : Proceedings of the 15th International Conference, 21–24 Sept. 2021, Minsk, Belarus. – Minsk : UIIP NASB, 2021. – 246 p. – ISBN 978-985-7198-07-8.

- Multiple people processing.

A. Image preprocessing

The first step of the algorithm is the analysis of the image by a convolutional network consisting of the first 10 layers of the VGG-19 network [1] and configured directly to generate characteristic maps, which, as a result of post-processing, send a set of these maps to the second step of the algorithm. The input is a color RGB image with size w×h, where w is width and h is height.

B. Simultaneous body parts detection and association

The neural network simultaneously predicts a set of probability maps $S_j \in R^{w\times h}, j \in \{1 \dots J\}$ and a set of two-dimensional compatibility vector fields L, which store the degree of compatibility between limbs. The set $L = (L_1, L_2, \dots, L_c)$ has vector fields C, where $L_c \in R^{w\times h\times 2}$, $c \in \{1 \dots C\}$. Each image position in L_c is encoded by a 2D vector.

The network is divided into two branches: the upper branch, predicts probability maps, and the lower branch, predicts the compatibility fields. Every branch has an iterative predictive architecture. An iteration stage, following Weil's method [2], refines the network predictions with intermediate control after each stage $t \in \{1 \dots T\}$.

At the first stage (t=1), the network creates a set of probability maps $S^1 = p^1(F)$ and a set of compatibility fields of parts $L^1 = b^1(F)$, where p^1 and b^1 are convolutional neural networks. At each stage t, the previous stage predictions S^t and L^t are combined with the initial characteristics F to obtain refinements of the network forecasts:

$$S^t = p^t(F, S^{t-1}, L^{t-1}), \forall t \geq 2,$$

$$L^t = b^t(F, S^{t-1}, L^{t-1}), \forall t \geq 2,$$

where p^t and b^t – convolutional neural networks outputs at stage t

We use an error L^2 between network predictions and known correct maps and fields. Here we spatially weigh the error functions to get rid of the problem that some datasets do not fully label all people

$$f_S^t = \sum_{j=1}^{J}\sum_{p} W(p) * \left\|S_j^t(p) - S_j^*(p)\right\|_2^2 ,$$

$$f_L^t = \sum_{c=1}^{C}\sum_{p} W(p) * ||S_c^t(p) - S_c^*(p)||_2^2,$$

$$f = \sum_{t=1}^{T}(f_S^t + f_L^t),$$

where S_j is the probability map, L_c is the vector compatibility field, W is the binary mask, p is pixel of image. If $W(p) = 0$ there is no characteristic information on the p.

The mask is used to avoid errors in the validity of positive predictions during training. Intermediate control at each stage solves the vanishing gradient problem by periodically replenishing the gradient.

1) Building Probability Maps to Detect Body Parts

To evaluate f_s during training, we generate probability maps from two-dimensional key points. Each probability map is a two-dimensional representation of what happens to a specific body part for each pixel. Ideally, if there is one person in the image, only one peak should be present in each probability map if the corresponding part is visible; if more than one person occurs, there must be a peak corresponding to each visible part j for each person.

First, we create individual probability maps, for each person. The x_j is the position of body part j, $j \in R^2$, for person k, $k \in R^2$. The value of the probability map at the point p $\in R^2$ is determined by the formula:

$$S_{j,k}^* = e^{-\frac{\left\|p - x_{j,k}\right\|_2^2}{\sigma^2}},$$

where σ corresponds to control of the concentration of the peak region.

The probability map to be predicted by the network is a collection of individual probability maps for every human's pose:

$$S_j^*(p) = max_k\left(S_{j,k}^*(p)\right),$$

We accept the maximum of all maps instead of the average to better demonstrate the accuracy of the peaks in occlusion. As result the probability maps predict positions of candidates for body parts.

2) Compatibility Vector Fields Calculation

A set of detected body parts is processed to form the poses of an unknown number of people. To do this, It is need a reliable measure of compatibility for every pair of detected body parts. One of the possible ways to measure compatibility is to introduce an additional point - the middle between each pair of parts and check its belonging to the limb. However, when people come together, these midpoints are likely to give false associations. These false associations arise from two limitations:

- This method encodes only the position, not the orientation of each limb.
- It reduces the support area of the limb to one point.

Pattern Recognition and Information Processing (PRIP'2021) : Proceedings of the 15th International Conference, 21–24 Sept. 2021, Minsk, Belarus. – Minsk : UIIP NASB, 2021. – 246 p. – ISBN 978-985-7198-07-8.

To address these limitations, a new representation, called part compatibility fields, is created. It retains both location information and orientation throughout the support area of the limb. The compatibility (affinity) of the parts is a two-dimensional vector field for each limb. For each pixel in an area belonging to a specific limb, the 2D vector encodes a direction from one part of the limb to another. Each type of limb has a corresponding field of compatibility connecting its body parts into a limb.

For evaluation f_L during training, we determine the vector compatibility field at point on the image p defined by formula:

$$L^*_{c,k}(p) = \begin{cases} v \text{ if } p \text{ lies on the limb } c,k \\ 0 \; otherwise \end{cases}$$

Let $x_{j1,k}$ and $x_{j2,k}$ be the positions of the joints of body parts $j1$ and $j2$ for a person k. If point p lies on a limb, the value of $L^*_{c,k}(p)$ is a unit vector $\boldsymbol{v}$ that points from $j1$ and $j2$ and shows the direction of the limb. The vector is equal to zero for all other points t:

$$\boldsymbol{v} = \frac{(x_{j1,k} - x_{j2,k})}{\left\|x_{j1,k} - x_{j2,k}\right\|_2}.$$

The set of points is defined as the number within the segment, those points p for which inequalities:

$$0 \leq \boldsymbol{v} * (p - x_{j1,k}) \leq l_{c,k},$$

$$\left|v_\perp * (p - x_{j1,2k})\right| \leq \sigma_i,$$

where σ_i is limb width, the distance in pixels, $l_{c,k}$ is a limb length, $k = \left\|x_{j1,k} - x_{j2,k}\right\|_2$, and $v_\perp$ is a perpendicular to $\boldsymbol{v}$ vector (Fig. 1).

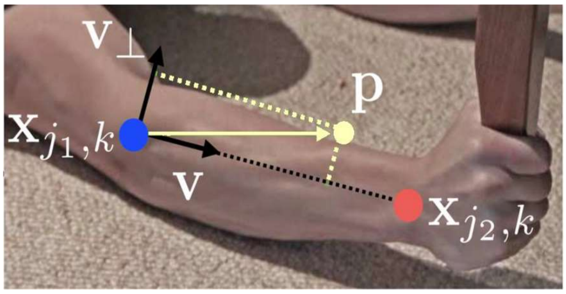

Fig. 1. Posipions of vectors for body part

The general affinity vector field $L^*_c(p)$ averages the fields $L^*_{c,k}(p)$ of all people on the image and is given by formula:

$$L^*_c(p) = \frac{1}{n_c(p)} \sum_k L^*_{c,k}(p),$$

where $n_c(p)$ is the number of nonzero vectors at point p for all k people (i.e., the average value in pixels where individual limbs for coordinate of people overlap).

For testing, we use measurement of the relationship between the identified candidates by calculating the sum:

$$E = \sum_{u=0}^{1} L_c\big(p(u)\big) * \frac{d_{j1} - d_{j2}}{\left\|d_{j1} - d_{j2}\right\|_2},$$

were $p(u)$ function that is calculated over the corresponding vector field, along the segment connecting the locations of the candidates:

$$p(u) = (1-u)d_{j1} + u \cdot d_{j2}.$$

In other words, we measure the alignment of the predicted field with the limb, which will be formed by connecting the detected body parts. In particular, for candidates d_{j1}, d_{j2} we project the affinity field of the predicted portion L_c along the line segment to measure the accuracy of their relationship.

In practice, we approximate the integral by sampling and summing uniformly distributed u values.

C. Multiple people processing

At this stage, probability maps and affinity fields are analyzed to output two-dimensional key points for all people in the image.

The algorithm does not perform the maximum suppression on the probability maps to obtain a discrete set of candidates for the next processing steps. Due to the analysis of all people at once or inaccuracy in image processing, several candidates may arise at once for each part of the body. These candidates constitute a large set of possible limbs. Every limb is controlled using the sum E.

The optimal selection problem corresponds to the K-dimensional matching problem, which is known to be NP-hard. This seems to be a rather expensive relaxation that consistently determines high-quality matching of parts in the limb. This is because the pairwise associative evaluation implicitly encodes the global context due to the high sensitivity of the compatibility fields.

First, we get a set of candidates D_J for several people, where $D_j=\{d^m_j$, for $j\in[1,...,J]$, $m\in[1,...,N_j]\}$, N_j is the number of candidates for part j and $d^m_j \in R^2$ is the location of the m-candidate for body part j. These candidates still need to be connected to other body parts from the same person. In other words, we need to find candidate pairs that are actually a connected limb. For this, a variable $z^{m,n}_{j1,j2} \in [0,1]$ is a flag indicating whether the two candidates are connected to each other. The goal is to find the optimal assignment for the set of all possible connections:

Pattern Recognition and Information Processing (PRIP'2021) : Proceedings of the 15th International Conference, 21–24 Sept. 2021, Minsk, Belarus. – Minsk : UIIP NASB, 2021. – 246 p. – ISBN 978-985-7198-07-8.

$$Z = \{z_{j1,j2}^{m,n}: for\, j1, j2\ \in [1, \dots, J], m \in [1, \dots, N_{j1}], n \in [1, \dots, N_{j2}]\}.$$

If we consider one pair of parts j_1, j_2 (for example, the neck and the right thigh) for the limb c, then the search for the optimal match is reduced to the problem of matching bipartite graphs with maximum weight [3]. In this problem, the nodes of the graph are the candidates for detecting a body part D_{j1} and D_{j2} and the edges are all possible connections between pairs of candidates. In addition, each edge has its own weight. A concordance in a bipartite graph is a subset of edges selected in such a way that no two edges separate a node. Our goal is to find a match Z with the maximum weight for the selected edges:

$$max_{Z_c} E_c = max_{Z_c} \sum\nolimits_{m \in D_{j1}} \sum\nolimits_{n \in D_{j2}} E_{m,n} z_{j1,j2}^{m,n},$$
$$\forall\, m \in D_{j1}, \sum\nolimits_{n \in D_{j2}} z_{j1,j2}^{m,n}\ \leq 1\square$$
$$\forall\, n \in D_{j2}, \sum\nolimits_{m \in D_{j1}} z_{j1,j2}^{m,n}\ \leq 1\square$$

Where E_c is the total weight of the correspondence for the part type C, Z_c is the subset Z for the part type C, $E_{m,n}$ is the correspondence between the parts $d^m{}_{j1}$ and $d^n{}_{j2}$.

The equations stipulate that two limbs of the same type do not share one part. We can use the Hungarian algorithm [4] to get the optimal match.

When it comes to finding the whole pose of many people, the Z definition is a K-dimensional matching problem. This problem is NP hard and there are many relaxations. We add two more relaxations to optimize this algorithm:

- The minimum number of edges is chosen to obtain the skeleton of a human pose, rather than using the full graph.
- The matching task is decomposed into a set of bidirectional matching subtasks and independently determines the matching in adjacent tree nodes:

$$max_Z E = \sum_{c=1}^{C} max_{Z_c} E_c \,.$$

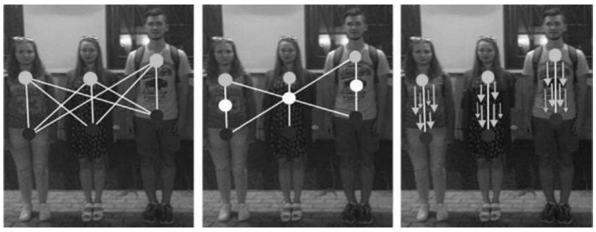

Fig. 2. Joints of body parts for every person on image

Therefore, candidates for limb connection are obtained independently of each other. These candidates are then assembled into limbs and combined into full body poses for multiple people (Fig. 2). This optimization scheme for a tree structure is an order of magnitude faster than optimization for a fully coupled schedule [5, 6].

The result of the algorithm is a full-size virtual human skeleton, displaying the positioning of his anatomical body parts, as well as the position of the whole body.

IV. Solution testing and results

The algorithm is evaluated according to two criteria:

- MPII multiuser dataset,
- COCO milestone dataset.

These two datasets contain images with different scenarios that contain many real-world problems such as large numbers of people, zooming, occlusion, and contact. Our approach set a new record for the speed of work on the COCO dataset [7] and significantly exceeds the previous modern results on MPII data [8] for several people. The table below shows some of the qualitative results of the algorithm.

A. Test Results on MPII Multi-Person Dataset

For comparison on the MPII dataset, we use the toolkit [32] to measure the mean accuracy (mAP) of all body parts based on the PCKh threshold. Figure 3.1 compares mAP performance between our method and other approaches. First, over a subset of 288 test images, and then over the entire MPI test suite. Besides the mAP score, we compare the average processing and optimization times on the image.

For a subset of 288 images, our method outperforms previous modern bottom-up methods by 8.5% mAP. It is noteworthy that our inference time is 6 orders of magnitude less than competing algorithms. Section 3.3 analyzes the runtime in more detail.

For the entire set of MPII tests, our method already surpasses the previous modern algorithms by a large margin, that is, by 13% mAP. Using 3-scale search (x0.7, x1 and x1.3) further increases performance to 75.6% mAP. Comparing mAP with previous bottom-up approaches shows us how effective the representation of functions, PAF, are for linking body parts. Based on a tree structure, our expensive processing method provides better accuracy than a graph optimization formula based on a fully coupled graph structure [32, 33].

B. Test results on COCO Keypoints Challenge dataset

The COCO training set consists of over 100,000 people with over 1 million major key points (body parts). The test suite contains subsets of "test-challenge", "test-dev" and "test-standard", which have approximately 20K images.

Algorithm	***AP***	***AP-50***	***AP-75***	***AP-M***	***AP-L***
Current	**60.5**	**83.4**	**66.4**	55.1	**68.1**
G-RMI	59.8	81.0	65.1	**56.7**	66.7
DL-61	53.3	75.1	48.5	55.5	54.8
R4D	49.7	74.3	54.5	45.6	55.6

The COCO score determines the comparability of the recognized points (OKS) and uses the average accuracy (AP) at more than 10 OKS thresholds as the main scoring criterion. OKS is calculated based on the size of the person and the distance between the predicted points and the GT points. Fig. 3.2 shows the results from the best teams for a task. It should be noted that our method has lower accuracy than top-down methods for people with smaller scales (APM) [16]. The reason is that our method has to deal with a much wider range of scales covered by all the people in the image in one frame. Top-down methods, on the other hand, can scale each deterministic region to a larger size and thus reduce the error at smaller scales.

C. Realtime Tests

To analyze the performance of our method, we collect videos from different numbers of people. The original frame size is 1080×1920, which we change to 368×654 during testing to match the GPU memory. Runtime analysis is performed on a laptop with a single NVIDIA GeForce GTX-1080 GPU.

In Fig. 3 we are using person detection and CPM for one person as opposed to top-down approaches, where the execution time is roughly proportional to the number of people in the image. The time it takes to complete our bottom-up approach grows relatively slowly as the number of people increases. The lead time consists of two main parts:

CNN processing time with O (1) complexity, which is constant with different numbers of people

Time of distribution of parts between people, the complexity of which is O (), where n is the number of people.

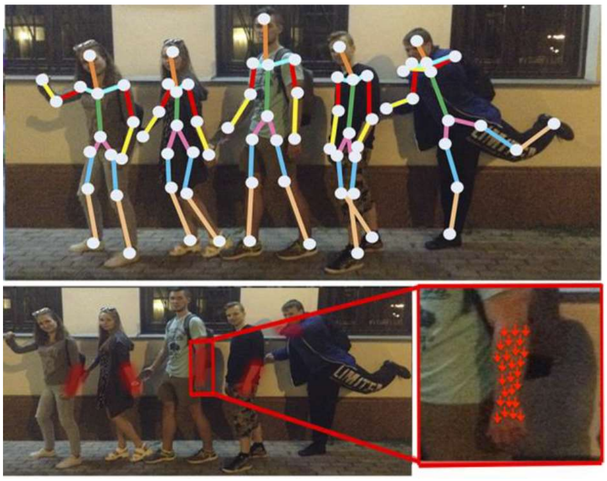

Fig. 3. Results detection of positions of body parts and intermediate vectors contruction

The parsing time does not have a significant impact on the overall execution time, as it is two orders of magnitude less than the processing time of CNN, for example, for 9 people, parsing takes 0.58 ms and CNN takes 99.6 ms. Our method achieved 8.8 frames per second for videos with 19 people.

ACKNOWLEDGMENT

This research was partly supported under BRFFR projects (F20KIGT-006 and F21UKR-001).

REFERENCES

[1] M. Sun, S. Savarese, Articulated part-based model for joint object detection and pose estimation, ICCV, 2011.

[2] D. B. West, Introduction to graph theory, vol. 2, Prentice hall Upper Saddle River, 2001.

[3] Y. Yang, D. Ramanan, Articulated human detection with flexible mixtures of parts, TPAMI, 2013.

[4] A. Newell, K. Yang, J. Deng, Stacked hourglass networks for human pose estimation, ECCV, 2016.

[5] L. Pishchulin, E. Insafutdinov, S. Tang, B. Andres, M. Andriluka, P. Gehler, and B. Schiele, Deepcut: Joint subset partition and labeling for multi person pose estimation, CVPR, 2016.

[6] E. Insafutdinov, L. Pishchulin, B. Andres, M. Andriluka, B. Schiele, Deepercut: A deeper, stronger, and faster multi-person pose estimation model, ECCV, 2016.

[7] T. Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollar, C. L. Zitnick, Microsoft COCO: common objects in context, ECCV, 2014.

[8] X. Chen, A. Yuille, Articulated pose estimation by a graphical model with image dependent pairwise relations, NIPS, 2014.

Pattern Recognition and Information Processing (PRIP'2021) : Proceedings of the 15th International Conference, 21–24 Sept. 2021, Minsk, Belarus. – Minsk : UIIP NASB, 2021. – 246 p. – ISBN 978-985-7198-07-8.