An Evaluation of Pose Estimation in Video of Traditional Martial Arts Presentation

Nguyen Tuong Thanh[1], Le Van Hung[2], Pham Thanh Cong[1]

[1] School of Electronics and Telecommunications, Hanoi University Science and Technology, Vietnam

[2] Tan Trao University, Vietnam

Correspondence: Le Van Hung, van-hung.le@mica.edu.vn

Communication: received 20 May 2019, revised 8 September 2019, accepted 11 September 2019

Digital Object Identifier: 10.32913/mic-ict-research.v2019.n2.864

The Editor coordinating the review of this article and deciding to accept it was Dr. Le Vu Ha

Abstract: Preserving, maintaining, and teaching traditional martial arts are very important activities in social life. That helps individuals preserve national culture, exercise, and practice self-defense. However, traditional martial arts have many different postures as well as varied movements of the body and body parts. The problem of estimating the actions of human body still has many challenges, such as accuracy, obscurity, and so forth. This paper begins with a review of several methods of 2-D human pose estimation on the RGB images, in which the methods of using the Convolutional Neural Network (CNN) models have outstanding advantages in terms of processing time and accuracy. In this work we built a small dataset and used CNN for estimating keypoints and joints of actions in traditional martial arts videos. Next we applied the measurements (length of joints, deviation angle of joints, and deviation of keypoints) for evaluating pose estimation in 2-D and 3-D spaces. The estimator was trained on the classic MSCOCO Keypoints Challenge dataset, the results were evaluated on a well-known dataset of Martial Arts, Dancing, and Sports dataset. The results were quantitatively evaluated and reported in this paper.

Keywords: *Estimation of keypoints, pose estimation, deep learning, skeleton, conserving and teaching traditional martial arts.*

I. Introduction

Estimating and predicting actions of human body are well-studied problems in the robotics and computer vision community [1]. Application domains include social safety, preservation of cultural identity values (conserving and maintaining traditional martial arts and national dance songs), production of toys and games, interaction with intelligent robots, sports analysis (tactical analysis in sports such as football, tennis, badminton, etc.), health protection (detection of falling events in hospital for the elderly), etc. Solving these problems can be based on a set of methods such as analyzing people in the images, locating people in the images, locating keypoints on human bodies, and identifying joints (skeleton) from the points featuring their body.

The problem of estimating the skeleton from the image of a person is usually based on color images, depth images, or the contexts of objects and actions [1]. The above systems often use color image information, depth information [2], or skeleton [3] obtained from different types of sensors. In particular, the Microsoft (MS) Kinect sensor version 1 (v1) is a common and cheap sensor that can collect several types of information such as color, depth, skeleton, and acceleration vector [4].

Color is the most common type of information obtained from cameras/sensors. Changes of appearance and posture of the human body structure in the image create a set of characteristics of deformation part model (DPM). That makes it difficult to estimate the shape and joints of the human body. The transformation of a complex human body is made up of changes in human body parts, which can be common transformations such as translation, rotation, and resizing. Previous studies often train DPM feature sets for detecting, recognizing, and estimating human postures and poses in images [5–7].

Recently, human pose estimation is still very challenging [3] in such terms as processing time and accuracy [8], especially for 3-D human pose estimation performed on datasets which have many occlusions [9]. Currently, with strong development of deep learning for detection, recognition, and human action estimation, it has become a good approach for solving these problems. There are many proposed Convolutional Neural Networks (CNNs) which achieved very good results in detecting and recognizing objects, such as Fast R-CNN [10], Faster R-CNN [11], and YOLO [12, 13]. Recently, there have been many studies of skeleton estimation on images using CNN models, such as [14–16].

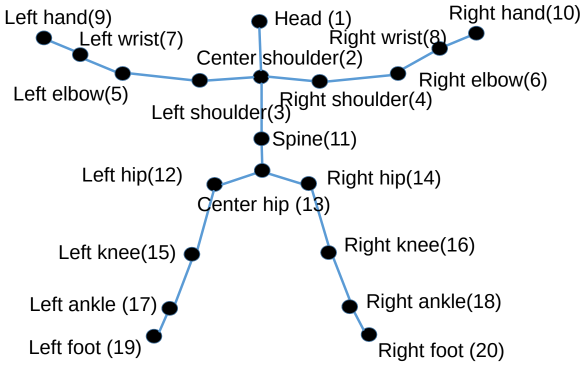

Figure 1. Keypoints on the human body and the labels.

In this work we only use the CNN model which was proposed and trained as [17] to estimate and predict the actions of people in videos of instructors and practitioners performing martial arts. This approach is based on the model that is trained from the confidence maps of feature vectors [18] which are extracted from the training images. The trained model estimates the keypoints on the human skeleton model as seen in Figure 1. In particular, this approach can estimate the posture of people based on the skeleton in case of being obscured.

Data obtained from the MS Kinect sensor includes color images, depth images, and correspondence. The first two types of data are calibrated to a center based on the approach and the intrinsic matrix of the MS Kinect sensor v1 proposed by Nicolas *et al.* [19]. Each frame builds a scene in 3-D environment. The estimated results of keypoints and joints are also transferred to 3-D space. It is then possible to build a martial arts teaching application in a more intuitive way.

The main contributions of this work include: (i) using a CNN model for 2-D human pose estimation on RGB images, achieving outstanding advantages in terms of processing time and accuracy; (ii) a small dataset of traditional martial arts videos and using the CNN-based pose estimation model [17] trained on the COCO [20] dataset to evaluate the skeleton estimations performed on the contributed dataset and the dataset of Zhang *et al.* [21]; and (iii) proposing measurements to evaluate human pose estimations in 2-D and 3-D spaces.

II. Related Works

In Vietnam [22, 23] as well as many countries in the world like China [24], Japan, and Thailand, there are many martial arts postures or martial arts that need to be preserved and passed down to posterity. Conservation and storage in the era of technology can be done in many different ways. An intuitive approach is to save the joints in the skeleton model of a martial arts instructor. Data obtained from MS Kinect sensor v1 usually contains a lot of noise and is lost when being obscured, especially skeleton data of people. The skeleton data is important and presents the human pose in a video action.

In the past, studies often looked into deformation part model features to address the problem of skeleton estimation on images. Felzenszwalb *et al.* [5] proposed a method for training a multiscale deformation part model (DPM) for object detection on images. In a partial deformation model approach, the human body is represented as a star-shaped structure, consisting of a root filter, a set of part detectors, and a partial deformation model. In the DPM model, deformation refers to relative positions of the body parts. An SVM (Support Vector Machine) classifier is trained on extracted features to predict the positions of the human body parts. Sun *et al.* [6] proposed a model based on the Articulated Part-based Model (APM) to detect parts of the human body and estimate the posture of the person. The APM model represents an object as a collection of parts at a level of details in the range from coarse to smooth, in which parts at all levels are connected to a coarser level through a parent-child relationship. Pishchulin *et al.* [25] as well as Andriluka [26] used the method of dividing the human body into parts and training the model on the parts for the estimation of one's body pose. Andriluka *et al.* [26] used AdaBoost for predicting the posture of the person. Umer *et al.* [27] used Regression Forests to estimate the direction of users on depth images obtained from MS Kinect sensor v2. The model estimation was performed on the parts of the labeled person, with 1000 sample position patterns on depth images. However, the highest average accuracy was just 35.77%.

Recently, with the strong development of deep learning, the estimation of keypoints on human body is often done by CNN models. Daniil *et al.* [14] introduced a new CNN model for learning the features on a keypoint dataset: the location of keypoints and the relationship between pairs of points on the human body. This new network is based on the OpenPose toolkit [17] and training can be done without GPU (CPU only). In particular, the CNN model of this study is trained and evaluated on the COCO 2016 key-point challenge dataset [20]. This is a huge dataset of labels containing images of over 150 thousands people with 1.7 million labels of keypoints. Kyle *et al.* [15] used a CNN model to learn from the data of the keypoints of the human body that were marked and extracted from the connection data when projecting two cameras into people. The result was then projected into 3-D space and then the least squared distance algorithm was used to evaluate the

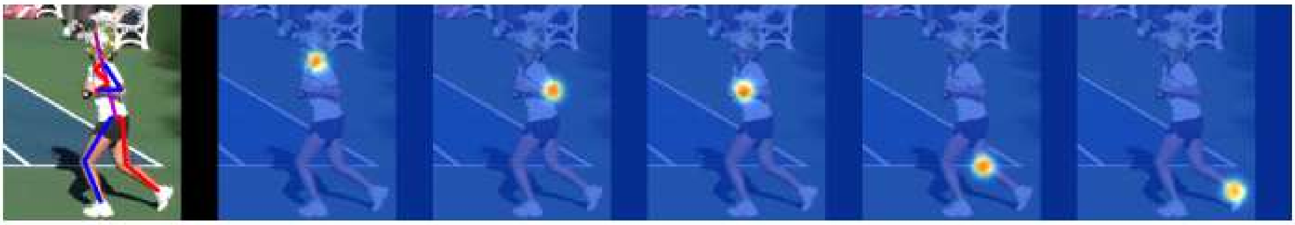

Figure 2. Illustration of heatmaps predicted from human body image. Therein, each heatmap is a candidate prediction of keypoint locations (x, y) [28].

obtained estimates. Cao *et al.* [18] used a CNN model to learn the positions of keypoints on the human body and the geometric transformations of the lines connecting the keypoint pairs with the above connected human body. Evaluations were conducted on two classic datasets, the MPII [29] and the COCO [20]. In particular, the COCO dataset of keypoints [30, 31] has been developed for many years. MPII and COCO datasets contain images of hundreds of thousands of people and have been used in many challenges/competitions on human activity estimation.

Toshev *et al.* [32] estimated human posture and skeleton, considering the human skeleton as a CNN-based regression. The authors also used a sequence of regression variables to correct the posture and skeleton estimations to get a better estimation. It is important that this method is based on a completed shape of posture and skeleton. When the joints are obstructed, they can be estimated from the completed posture and the skeleton structure. The model in this study is trained on the AlexNet backend network of seven layers, in which the final layer is used to complete the trained model and the target output values of the regression, the number of target values was about 2000 in term of joint coordinates.

Tompson *et al.* [28] created heatmaps by simultaneously running an image through many different resolutions to collect multi-resolution features at the same time. The output is a discrete heatmap instead of continuous regression. A heatmap predicts the probability of joints at each pixel, in which each heatmap area is a candidate of the location of keypoints on the human body (heatmap areas are created as shown in Figure 2).

In addition to the training method and the prediction of heatmaps, Wei *et al.* [33] proposed a network that trains through many phases on the characteristic set of images. They provided a sequential predictive framework focusing on training highly predictive models. Output heatmaps of the previous stage are used as input of the later stages, with which the best accuracy obtained by this model on the MPII dataset [34] was 87.95%.

Andriluka *et al.* [35] published a dataset that is structured and organized similarly to the classic datasets. These benchmark datasets provide training sets, validation sets, and testing sets to train and evaluate deep learning-based methods. In particular, they also established performance metrics for direct and fair comparison across numerous competing approaches. Girdhar *et al.* [16] proposed a 3-D human pose predictor by inflating the 2-D convolutions into 3-D [36] to extend the Mask RCNN [37] with spatio-temporal operations. This work used the Mask RCNN [37] for 2-D human pose estimation, the tracking process was the combination of 2-D human pose estimation results and temporal information. This method achieved high accuracies on the challenging PoseTrack benchmark dataset [35]. However, since this method must predict the box, the segmentation, and the keypoints, the processing time of testing is large (5 frames/s with 8 GPUs). Meanwhile, the method proposed by Cao *et al.* [18] is able to process about 10-15 frames/s with a 12 GB GPU.

III. Human Pose Estimation

The activity of the human body is detected and recognized as well as predicted and estimated based on body parts. 3-D human pose estimation employs either one of the two basic methods: (1) estimating the 3-D human pose from a single image (RGB or depth), and (2) estimating the 3-D human pose from an image sequence. There are many studies on 3-D human pose estimation that use the single-image method [9, 38, 39]. In these studies, the keypoints and joints are estimated on 2-D images and then mapped to the 3-D space. This model is often applied to estimating 3-D human pose, as shown in Figure 3 [40].

In this paper, to estimate the 2-D human pose, we use the method of Cao *et al.* [18]. The architecture of the CNN to train the model is shown in Figure 3. This CNN consists of two branches performing two different jobs. From the input data, a set **F** of feature maps is created from image analysis, these confidence maps and affinity fields are detected at the first stage.

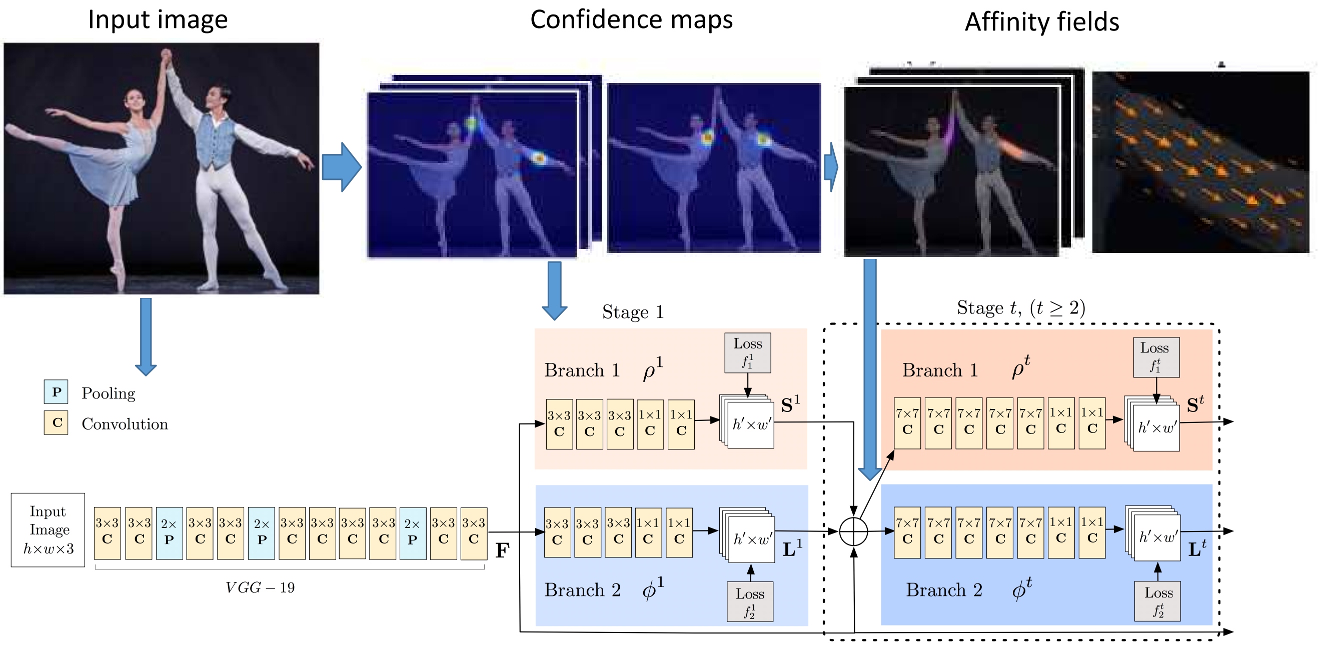

Figure 3. The architecture of the two-branch multi-stage CNN for training the estimation model [18].

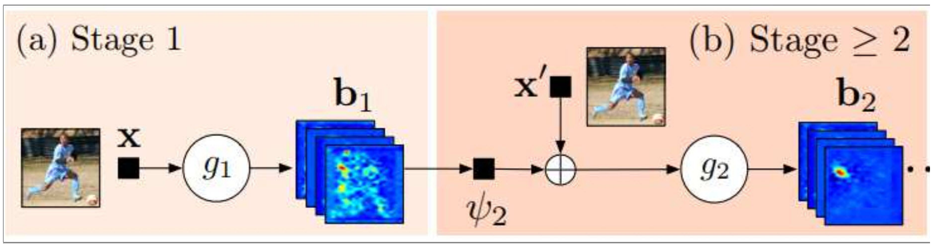

Figure 4. Illustration of the detailed model to predict heatmaps [33].

Details on Cao's model training and prediction (Figure 4) [18] are shown as follows. The input image at stage 1 is an RGB image which has a size of $h \times w$. Features extracted from convolutions with masks of sizes $9\times 9, 2\times 2, 5\times 5, \ldots$ for the training set $\mathbf{X}$ as shown in Figure 5. For each mask, there will be a trained model at each stage. As shown in Figure 4, models g_1 and g_2 at stages 1 and 2 will predict the heatmaps b_1 and b_2, respectively. In Figures 4 and 5, the Convolutional Pose Machines consist of at least 2 stages and the number of phases is a super parameter (usually 3 stages). The second stage takes the resulted heatmaps of the first stage as the input.

Therein, each heatmap indicates the location confidence of the keypoints as a function of (x, y). Keypoints on the training data are displayed on confidence maps as shown in Figure 4. These points are estimated by the trained model as the keypoints on input color images. The first branch (top branch) is used to estimate the keypoints, the second branch (bottom branch) is used to predict the affinity fields matching joints on people.

In addition, we also render a 3-D environment of each video's scene and project the results of 2-D human pose estimation into the 3-D space, based on the intrinsic parameter of the Kinect sensor v1, using the PCL library [41] and the OpenCV library [42] functions. The real coordinates (x_p, y_p, z_p) and color values of each pixel when projected from 2-D to 3-D space are calculated as in Eq. 1.

$$\begin{array}{rl} X_p & = \frac{(x_a - c_x) * depthvalue(x_a, y_a)}{f_x} \\ Y_p & = \frac{(y_a - c_y) * depthvalue(x_a, y_a)}{f_y} \\ Z_p & = depthvalue(x_a, y_a) \\ C(r, g, b) & = colorvalue(x_a, y_a) \end{array} \tag{1}$$

where $depthvalue(x_a, y_a)$ is the depth value of a pixel at (x_a, y_a) on the depth image, and $colorvalue(x_a, y_a)$ returns the RGB color values of that pixel on the color image.

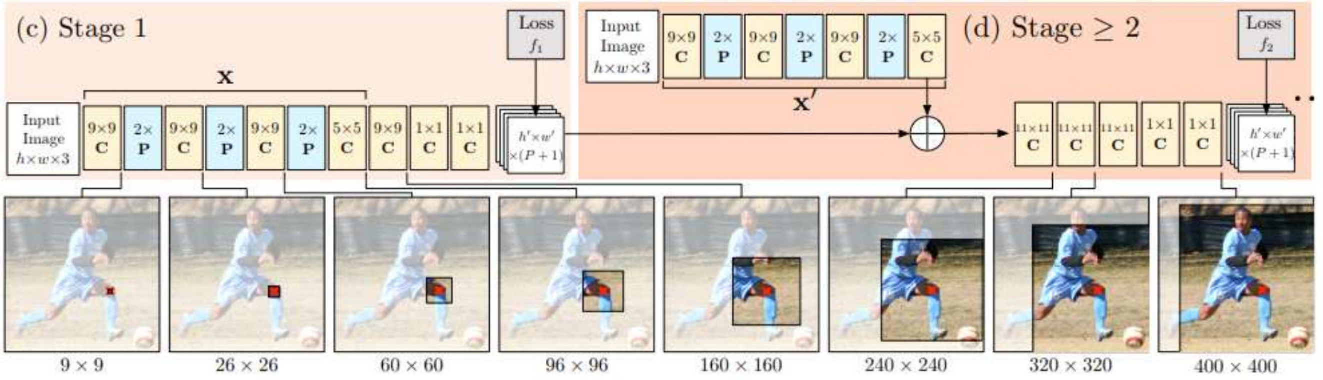

Figure 5. Illustration of the detailed model to extract features for training model and to predict heatmaps at each stage [33].

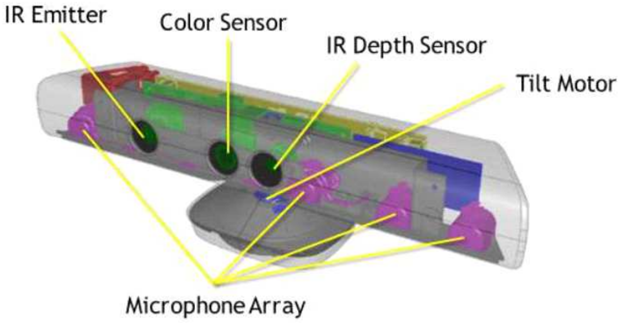

Figure 6. MS Kinect sensor v1.

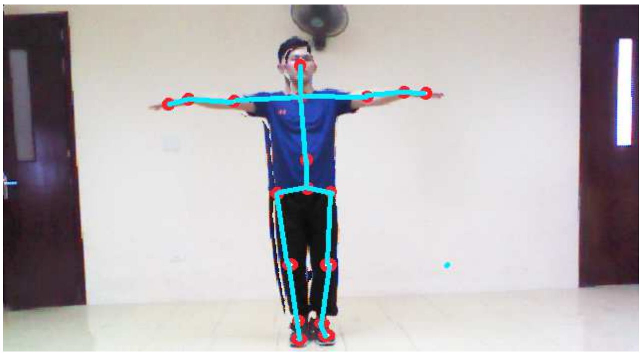

Figure 7. Illustration of the obtained image from MS Kinect sensor v1 and annotated keypoints of the skeleton model.

In regard to the process of combining color and depth information of a pixel to obtain a point in 3-D space, for cases where the depth values of pixels in the depth image are lost (value is zero), we use the average depth value of pixels in their 50×50 neighborhood.

In our experiments, results of 2-D human pose estimation are the keypoints $\{(x_e, y_e) | e \in \{1, 2, ..., 25\}\}$. The joints are then joined according to a predefined order.

IV. Experimental Result

1. Data Collection

There are many different types of image sensors that can collect information about martial arts teaching and learning. The MS Kinect v1 sensor as seen in Figure 6 is the cheapest sensor today. This type of sensor can collect a lot of information such as color images, depth images, skeletons, acceleration vectors, sound, etc. From the collected data, it is possible to recreate the environment in 3-D space. However, in this work we only use color images captured by the MS Kinect v1 sensor.

To capture data from the sensor, the MS Kinect SDK 1.8 is used [43]. To perform data collection on computers, we use a data collection program developed at MICA Institute [44] with the support of the OpenCV 3.4 libraries [42]. Calibration is required in order to generate 3-D data from color and depth images. Particularly, we apply the calibration methods of Zhou *et al.* [45] and Jean *et al.* [46]. In these two calibration tools, the calibration matrix is used as follows:

$$H_m = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}, \tag{2}$$

where (c_x, c_y) is the principle point (usually the image center) and (f_x, f_y) is the focal length vector. The matrix H_m (in Nicolas *et al.* [19]) is given as follows:

$$H_m = \begin{bmatrix} 594.214 & 0 & 339.307 \\ 0 & 591.040 & 242.739 \\ 0 & 0 & 1 \end{bmatrix}. \tag{3}$$

In this work, we use two datasets for evaluating the model pose estimation [17]. The first dataset is collected from a

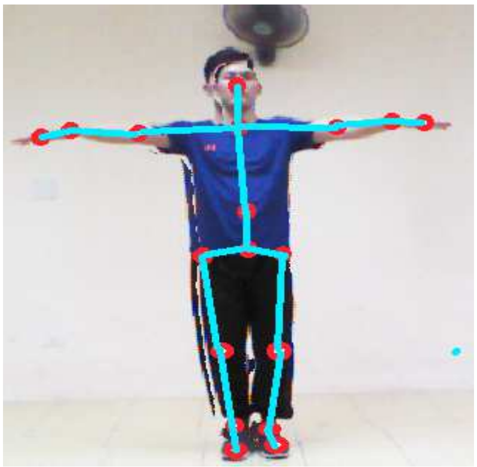

Figure 8. Illustrations on ground-truth for keypoints. Red points are keypoints on the human body. Blue segments show connections between parts of the human body.

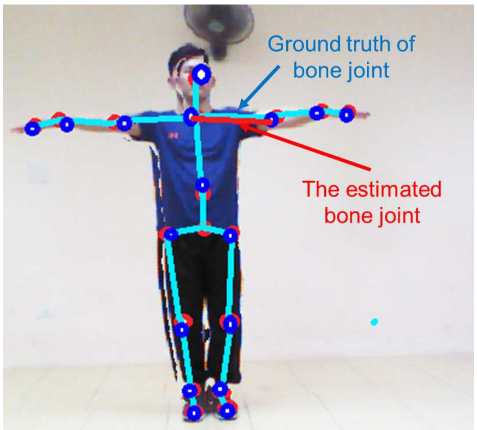

Figure 9. Illustration of the estimated results of the keypoints and joints.

MS Kinect v1 sensor, which can collect data at a rate of about 10 frames/s on a low-performance laptop. The MS Kinect sensor v1 is mounted on a fixed rack; the martial arts instructor presents in a 3 × 3m space as in Figure 10. It is called "VNMA - VietNam Martial Arts," captured in a martial arts class in Binh Dinh province, Vietnam. Binh Dinh martial art is one of the famous traditional martial arts of Vietnam.

The obtained images (color images and depth images) are 640×480 in pixels. The dataset consists of 14 videos of different postures, with the number of frames listed in Table I and illustrated in Figure 8. This dataset features a martial arts instructor with 14 different postures. The number of frames is the number of poses in each video. The ground-truths for keypoints are manually prepared, as illustrated in Figures 8 and 9. The ground-truth data in each image, which contains a single person, includes 18 keypoints.

TABLE I
NUMBER OF FRAMES IN MARTIAL ARTS POSTURES

Video	**1**	**2**	**3**	**4**	**5**	**6**	**7**
Number of frames	120	74	100	87	80	88	87
Video	**8**	**9**	**10**	**11**	**12**	**13**	**14**
Number of frames	74	71	90	100	97	65	68

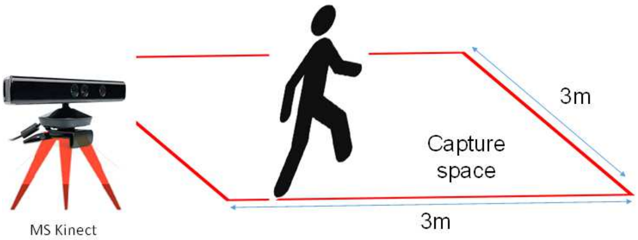

Figure 10. Illustration of MS Kinect v1 sensor settings and data collection space.

The second dataset called Martial Arts, Dancing, and Sports[21] consists of both multi-view RGB videos and depth videos. This dataset contains five action types: Tai-chi, Karate, Hip-hop dance, Jazz dance, and sports. The frame rate is 10 fps for Tai-chi and Karate, or 20 fps for jazz, hip-hop, and sports videos. The resolution of the images are 1024×768 in pixels. However, all the frames in this dataset are resized to 512 × 384 pixels. Ground-truth data was prepared for 3-D poses, using a MOCAP (MOtion CAPture) system [47] by Motion Analysis. Seven MOCAP cameras were placed on the walls around the capture space to record the positions of markers on the human body. The MOCAP system works at 60 fps. The dataset contains ground-truth data for 19 joints: neck, pelvis, left hip, left knee, left ankle left foot, right hip, right knee, right ankle right foot, left shoulder, left elbow, left wrist, left hand, right shoulder, right elbow, right wrist, right hand, and head.

We use only Tai-chi and Karate videos of color and depth images that contain 11200 frames. The camera calibration matrix of the capture system is given by

$$H_m = \begin{bmatrix} 331 & 0 & 254.097 \\ 0 & 331 & 180.032 \\ 0 & 0 & 1 \end{bmatrix}. \quad (4)$$

Figure 11 illustrates the point cloud data of a scene when a person presents Karate.

We use the 2016 MSCOCO Keypoints Challenge dataset [48] to train the model, because our collected dataset (VNMA - VietNam Martial Arts) is small and the MADS (Martial Arts, Dancing, and Sports) dataset provides

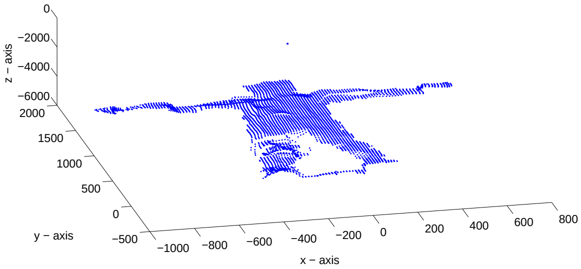

Figure 11. Illustration a point cloud data of a scene. The blue points represents the human body in the 3-D environment.

only 3-D ground-truth data. The training dataset consists of over 100K people and a million labeled keypoints. Each person is marked by 17 keypoints, which are sorted by the following order: "nose," "left_eye," "right_eye," "left_ear," "right_ear," "left_shoulder," "right_shoulder," "left_elbow," "right_elbow," "left_wrist," "right_wrist," "left_hip," "right_hip," "left_knee," "right_knee," "left_ankle," and "right_ankle." The model is trained for estimating and labeling these 17 keypoints on the human body.

A toolkit [49] is used for model training and testing. This toolkit is installed on Ubuntu 16.2 and supported by several libraries: OpenPose C++ library (for testing only) [17], Tensorflow version 1.8 [50], Pytorch [18, 51], Caffe2 [52], Chainer [53], MXnet [54], MatConvnet [55], and CNTK [56]. Readers are referred to [18, 48] for details. The training tool and the testing tool are written in Python/Matlab/C++ language and run on a workstation computer which has the following configurations: Intel (R) Xeon (R) CPU E5-2420 v2 @ 2.20 GHz 16 GB RAM and GTX 1080 Ti GPU with 12 GB RAM. The trained model is used to estimate 25 keypoints in the following order: "nose," "neck," "right shoulder," "right elbow," "right wrist," "left shoulder," "left elbow," "left wrist," "mid hip," "right hip," "right knee," "right ankle," "left hip," "left knee," "left ankle," " right eye," "left eye," "right ear," "left ear," "left big toe," "left small toe," "left heel," "right big toe," "right small toe," and "right heel."

The CNN training parameters are as follows[1]: size of the input images is $368 \times 368 \times 3$ (width x height x number of channels), $batchSize$ = 16, $stacks$ = 4, and the number of stages is 6 for pooling. Our collected VNMA dataset is still small, therefore we have not divided this dataset into training set, validation set, and testing set. This dataset is used only for testing.

2. Evaluation Method

As in [18], we use Object Keypoint Similarity (OKS) measure for evaluating the similarities between estimated and ground-truth joints. The formula for calculating OKS is given by [48]:

$$\text{OSK} = \frac{|G_{\text{ground}} - R_{\text{result}}|}{G_{\text{ground}}}, \tag{5}$$

where G_{ground} is the length of a ground-truth joint vector, R_{result} is the length of the corresponding estimated joint vector. The Average Precision (AP) is calculated based on the threshold value of 0.5 for OKS. Therein, if OKS > 0.5, meaning the difference is greater than 50% of length, then it is considered a false estimate, otherwise it is a true estimate.

We also propose another metric called the angle of deflection between an estimated joint $\mathbf{V}_E$ its corresponding ground-truth joint $\mathbf{V}_G$, which is the angle between the two vectors: $A = \arccos(\mathbf{V}_G, \mathbf{V}_E)$. If $A \leq 10^o$ it is considered a true estimate, otherwise it is a false estimate. Denote AD the ratio of true estimates over the total number of joints, and DP the average distance between estimated keypoints and ground-truth keypoints.

The average distance in 2-D space is given by

$$\text{DP}_{\text{2D}} = E\left[\sqrt{(x_g - x_e)^2 + (y_g - y_e)^2}\right], \tag{6}$$

where x_e and y_e are the coordinates of an estimated keypoint, while x_g and y_g are the coordinates of its corresponding ground-truth keypoint, E is the expectation operator.

The average distance in 3-D space is given by

$$\text{DP}_{\text{3D}} = E\left[\sqrt{(x_g - x_e)^2 + (y_g - y_e)^2 + (z_g - z_e)^2}\right], \tag{7}$$

where x_e, y_e, and z_e are the 3-D coordinates of an estimated keypoint, while x_g, y_g, and z_g are the 3-D coordinates of its corresponding ground-truth keypoint.

3. Results and Discussion

Performance of 2-D pose estimation on the VNMA dataset is presented in Table II. The average value of AP in Tabble II is 95.6%. For noisy data performance could be significantly lower, an example is video #4 (AP = 89.6%).

Pose estimation results are shown in Table II and Figure 12, in which 25 keypoints are estimated on the human body. However, the corresponding ground-truth data provides only 20 keypoints on each person, therefore the performance evaluation is performed on only 20 of the 25 estimated keypoints. It can be seen that the estimation results are highly accurate, although the model trained on the MSCOCO Keypoints Challenge dataset [48] and our test data contain a lot of noise. The average accuracy ratio

[1] Detailed parameters are shown in https://github.com/ZheC/Realtime_Multi-Person_Pose_Estimation/blob/master/training/example_proto/pose_train_test.prototxt.

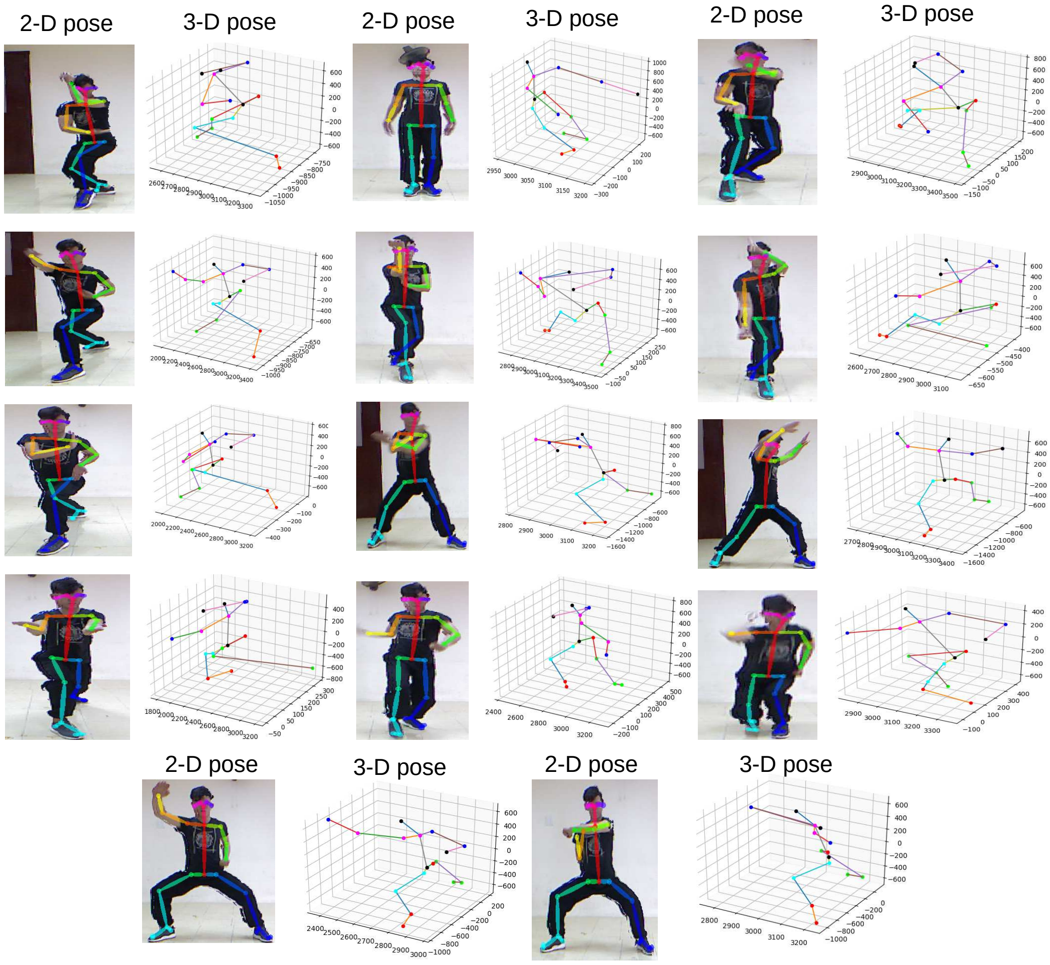

Figure 12. Results of joint estimation shown in 2-D and 3-D spaces.

AD, which is based on the angles of deflection between the estimated joints and the ground-truth, is 95.3% for this test dataset. The average value of DP_{2D} which is the deviation between estimated keypoints and the ground-truth in 2-D image space is 14.73 pixels[2]. Figure 12 shows estimated skeletons in 2-D and 3-D spaces, represented by 17 keypoints[3].

We also measure prediction probabilities in term of Intersection of Union (IOU) for all keypoints of three videos. These probabilities are represented as heatmaps produced by the CNN model, in which each heatmap area shows the confidence scores associated with candidate positions of a keypoint, as presented in Figures 4 and 5. Resulted probability distributions are shown in Figure 13. We can see that all three distributions are peeked within 0.8 to 0.9. That means the trained model in [17] has good predictability.

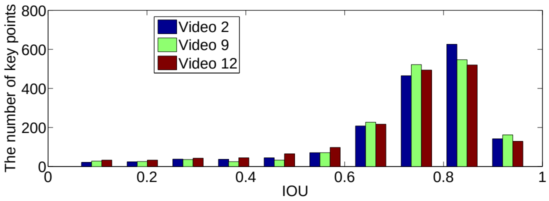

Figure 13. Probability distributions of the keypoint candidates predicted by the CNN model for three martial arts videos.

Performance of 3-D pose estimation on the MADS dataset is presented in Tables III and IV. From the figures in Table III, the average value of AP is 75.93% and average accuracy ratio AD is 79.62% and the estimated 2-D skeleton results are illustrated in Figure 14. Table IV compares performances of the TGP-KNN method [57] and the Pose toolkit [17] on the MADS dataset, using the

[2] Details of the estimated results are available at this link: https://drive.google.com/file/d/1BaFxUbOJvRll5tkgYQW98VOp6LC414Im/view?usp=sharing.

[3] The source code using OpenPose library is shared at the following link: https://drive.google.com/file/d/1OZO8m7TvtZUD55zf1TAU1HTWCk-u4pb8/view?usp=sharing.

TABLE II
AVERAGE PRECISIONS FOR ESTIMATED JOINTS (AP), AVERAGE RATIOS OF TRUE ESTIMATES FOR JOINTS (AD), AND AVERAGE DEVIATIONS OF ESTIMATED KEYPOINTS IN 2-D IMAGE SPACE (DP_{2D}) OBTAINED ON THE VIDEOS OF THE VNMA DATASET

Video	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**
AP (%)	95.4	93.7	96.2	89.6	96.1	92.8	97.4	98.8	96.9	94.5	96.9	96.2	95.7	98.2
AD (%)	93.7	94.6	92.8	90.9	95.3	94.6	95.8	97.6	97.8	95.1	97.0	95.8	96.3	96.9
DP_{2D} **(pixels)**	21.2	18.6	9.7	25.9	13.8	15.7	9.4	15.4	12.4	10.1	14.0	12.8	11.3	16.9

TABLE III
AVERAGE PRECISIONS FOR ESTIMATED JOINTS (AP) AND AVERAGE RATIOS OF TRUE ESTIMATES FOR JOINTS (AD) OBTAINED ON THE VIDEOS OF THE MADS DATASET

Video	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
AP (%)	71.20	70.53	72.64	60.53	80.50	73.22	81.95	91.31	55.19	86.65	79.98	87.46
AD (%)	80.45	75.83	80.76	62.46	78.77	78.93	89.13	87.65	59.84	86.46	86.94	88.2

TABLE IV
AVERAGE 3-D DEVIATIONS DP_{3D} (MM) BETWEEN ESTIMATED KEYPOINTS AND THE GROUND-TRUTH FOR VIDEOS OF THE MADS DATASET, USING TWO DIFFERENT METHODS

Video	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
TGP-KNN [57]	214.4	167.4	**246.7**	149.3	**119.5**	197.2	119.4	**114.2**	**214.5**	113.2	**98.4**	133.7
Pose toolkit [17]	**198.23**	**154.38**	252.3	**148.44**	120.34	**188.75**	**112.68**	116.3	**206.65**	**104.29**	102.34	**120.55**

deviations between estimated keypoints and the ground-truth in 3-D space. The TGP-KNN results are those reported in [21]. The processing time to estimate keypoints and joints is about 75 ms per frame.

Figure 15 illustrates the color point cloud of a scene in the VNMA dataset. Therein, the coordinate system (the x-axis is colored red, the y-axis is colored green, and the z-axis is colored blue) of the MS Kinect v1 sensor is shown in the original frame. Figure 16 illustrates 3-D joint estimation results from a scene. The human body is presented by the point cloud.

Figure 17 shows the estimated results in the 2-D space of the proposed dataset. This dataset were collected from a MS Kinect v1 sensor. However, in martial arts presentation, cases vary and the posture of the person must rotate in many different directions. Therefore, there are many cases where many actions are obscured. In order to build a conservation application for training martial arts and evaluating martial arts videos, the bone joints of these actions should be restored.

V. CONCLUSION AND FUTURE WORK

In this paper, we first reviewed some methods for 2-D human pose estimation on RGB images. It was proposed to use a CNN model for estimating keypoints to predict the actions of the martial arts instructor on a traditional martial arts video dataset that we contribute. Finally, we presented methods for evaluating the estimated keypoints and joints by 2-D and 3-D metrics. From the estimated joints human actions can be drawn about. Therefore, training martial arts by videos becomes easier and more explicit.

In martial arts videos, the actions (movements of body, arms, and legs) of a martial arts instructor are not always clear because there are many hidden joints. There are some cases where the joints are obscured in the videos that the model was unable to estimate. In the future, we will conduct studies to estimate obstructed joints. When there are sufficient joints, it is possible to build a visual martial arts teaching model and to evaluate the performance of traditional martial arts representations.

REFERENCES

[1] W. Gong, X. Zhang, J. Gonzàlez, A. Sobral, T. Bouwmans, C. Tu, and E. H. Zahzah, “Human Pose Estimation from Monocular Images: A Comprehensive Survey,” *Sensors (Basel, Switzerland)*, vol. 16, no. 12, pp. 1–39, 2016.

[2] M. Rantz, T. Banerjee, E. Cattoor, S. Scott, M. Skubic, and M. Popescu, “Automated fall detection with quality improvement "rewind" to reduce falls in hospital rooms,” *J Gerontol Nurs*, vol. 40, no. 1, pp. 13–17, 2014.

[3] R. IgualCarlos, M. Carlos, and I. Plaza, “Challenges, Issues and Trends in Fall Detection Systems,” *BioMedical Engineering OnLine*, vol. 12, no. 1, p. 147–158, 2013.

[4] J. Kramer, M. Parker, D. Castro, N. Burrus, and F. Echtler, “Hacking the Kinect,” *Apress*, 2012.

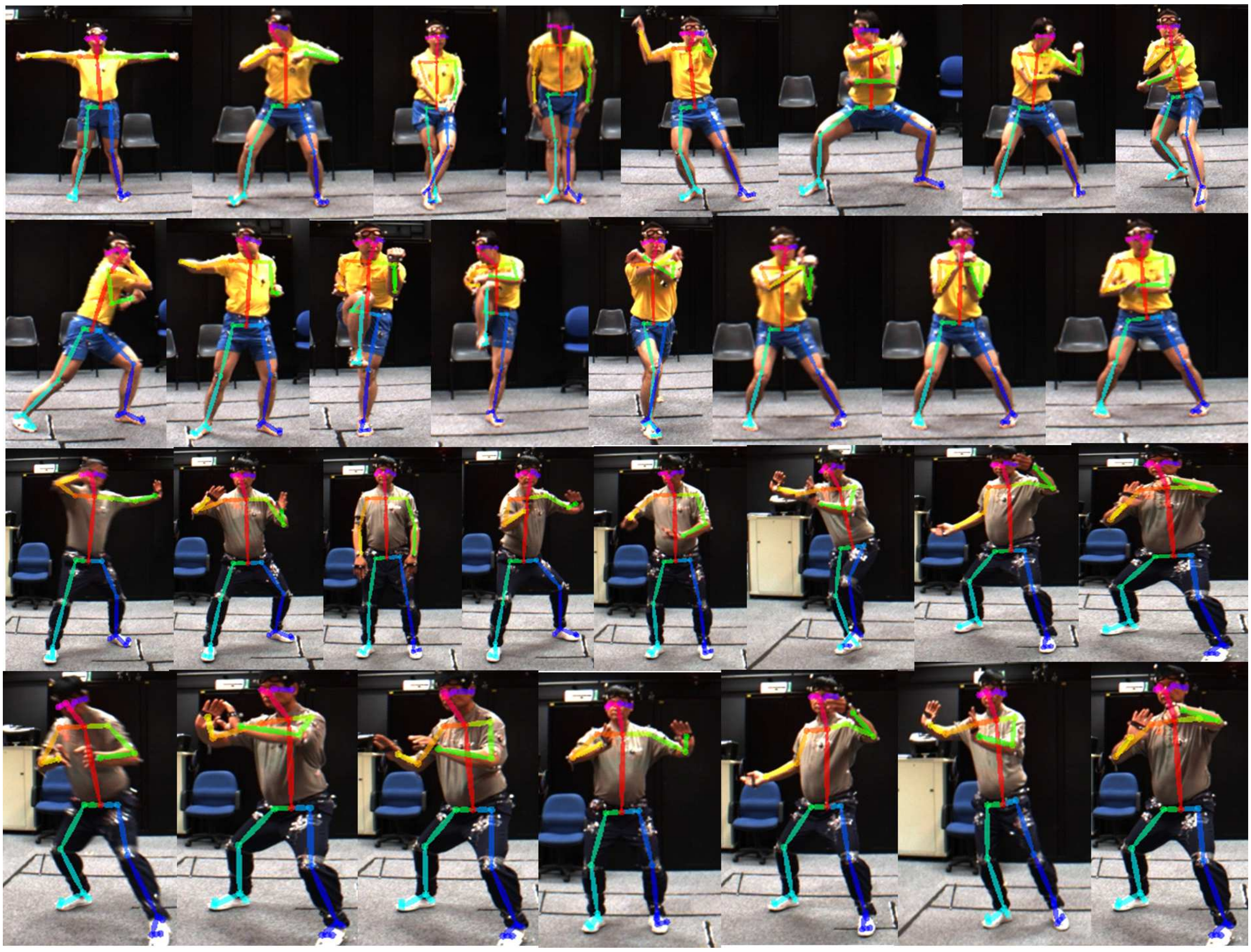

Figure 14. Estimated keypoints and joints on video frames showing chains of traditional martial arts actions.

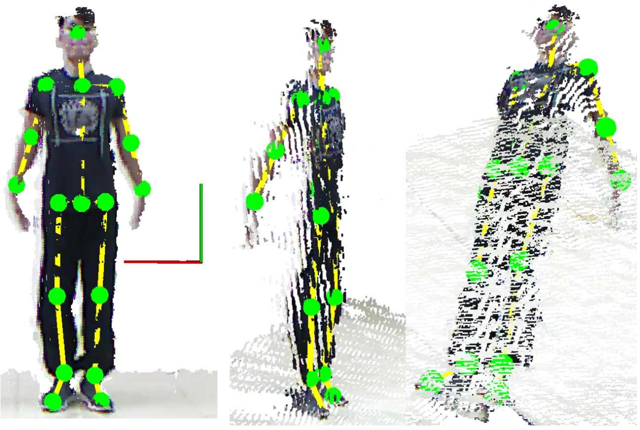

Figure 15. The color point cloud of a scene in a VNMA video, the estimated keypoints are colored green and the estimated joints are colored yellow.

[5] P. Felzenszwalb, D. Mcallester, and D. Ramanan, “A discriminatively trained, multiscale, deformable part model,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–8.

[6] M. Sun and S. Savarese, “Articulated part-based model for joint object detection and pose estimation,” in *IEEE International Conf. Computer Vision*, 2011, pp. 723–730.

[7] E. M. Berti, A. J. S. Salmerón, and C. R. Viala, “4-Dimensional deformation part model for pose estimation using Kalman filter constraints,” *International Journal of Advanced Robotic Systems*, vol. 14, no. 3, pp. 1–13, 2017.

[8] U. Rafi, J. Gall, and B. Leibe, “A semantic occlusion model for human pose estimation from a single depth image,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 67–74.

[9] I. Sarandi, T. Linder, K. O. Arras, and B. Leibe, “How robust is 3D human pose estimation to occlusion,” in *IEEE/RSJ International Conf. Intelligent Robots and Systems*, 2018.

[10] R. Girshick, “Fast R-CNN,” in *International Conference on Computer Vision*, 2015, pp. 1440–1448.

[11] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards real-time object detection with region proposal networks,” in *Advances in Neural Information Processing Systems 28*, 2015, pp. 91–99.

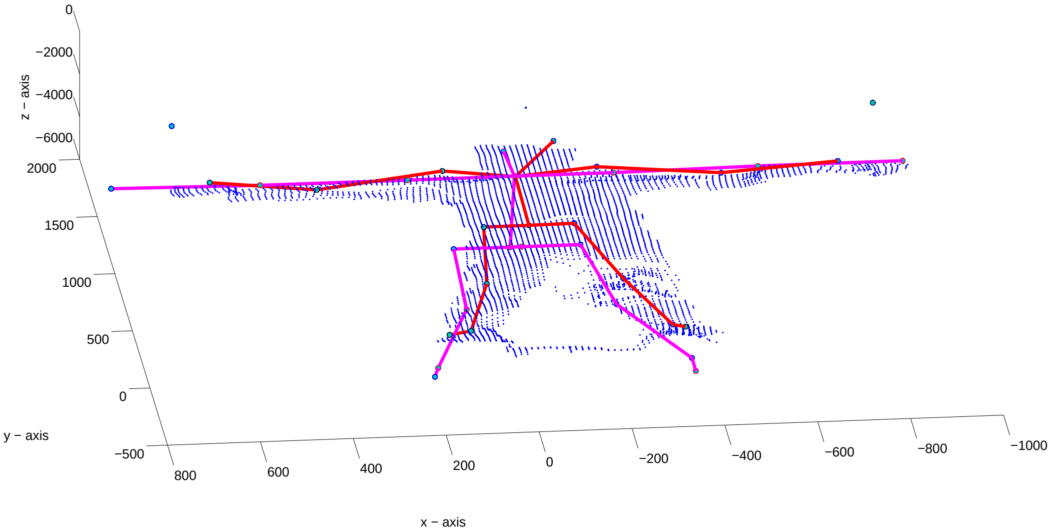

Figure 16. Estimated joints of the human body in 3-D space. The point cloud data of the human body is colored blue. Red segments are the ground-truth joints and magenta segments are the estimated joints.

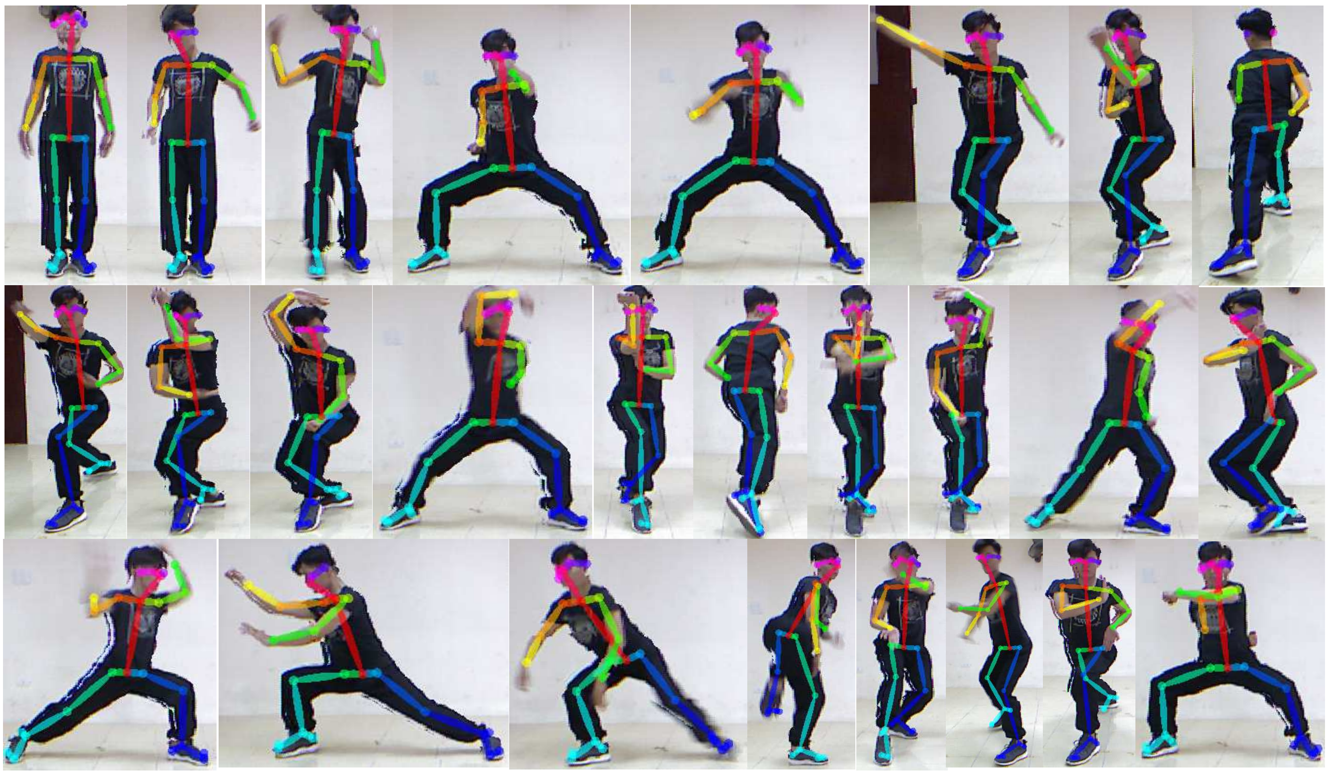

Figure 17. Estimated keypoints and joints on video frames showing a chain of martial art poses: body joints are colored red, left hand joints are colored green, right hand joints are colored yellow, left foot joints are colored blue, and right foot joints are colored cyan.

[12] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Comp. Vision and Pat. Recognition*, 2016, pp. 779–788.

[13] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Computer Vision and Pattern Recognition*, 2017, pp. 7263–7271.

[14] D. Osokin, "Real-time 2D multi-person pose estimation on CPU: Lightweight openpose," *ArXiv*, 2018.

[15] K. Brown, "Stereo human keypoint estimation," *Stanford University*, 2017.

[16] R. Girdhar, G. Gkioxari, L. Torresani, M. Paluri, and D. Tran, "Detect-and-track: Efficient pose estimation in videos," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 350–359.

[17] "OpenPose," [Accessed 23 April 2019]. [Online]. Available: https://github.com/CMU-Perceptual-Computing-Lab/openpose

[18] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2D pose estimation using part affinity field," 2017, pp. 7291–7299.

[19] B. Nicolas, "Calibrating the depth and color camera," 2018, [Accessed 10 January 2018]. [Online]. Available: http://nicolas.burrus.name/index.php/Research/KinectCalibration

[20] T.-Y. Lin, Y. Cui, G. Patterson, M. R. Ruggero, L. Bourdev, R. Girshick, and P. Dollar, "COCO 2016 Keypoint Detection Task," [Accessed 17 April 2019]. [Online]. Available: http://cocodataset.org/#keypoints-2016

[21] W. Zhang, Z. Liu, L. Zhou, H. Leung, and A. B. Chan, "Martial arts, dancing and sports dataset: a challenging stereo and multi-view dataset for 3D human pose estimation," *Image and Vision Computing*, vol. 61, pp. 22–39, 2017.

[22] T. B. Dinh, "Bao ton va phat huy vo co truyen Binh dinh: Tiep tuc ho tro cac vo duong tieu bieu," 2017, [Accessed 4 April 2019]. [Online]. Available: http://www.baobinhdinh.com.vn/viewer.aspx?macm=12&macmp=12&mabb=88043

[23] ——, "Ai ve Binh Dinh ma coi, con gai Binh Dinh bo roi di quyen," 2019, [Accessed 4 April 2019]. [Online]. Available: http://www.seagullhotel.com.vn/du-lich-binh-dinh/vo-co-truyen-binh-dinh-5

[24] Chinese, "Chinese Kung Fu (Martial Arts)," 2019, [Accessed 4 April 2019]. [Online]. Available: https://www.travelchinaguide.com/intro/martial_arts/

[25] L. Pishchulin, M. Andriluka, P. Gehler, and B. Schiele, "Strong appearance and expressive spatial models for human pose estimation," in *IEEE International Conference on Computer Vision*, 2013, pp. 3487–3494.

[26] M. Andriluka, S. Roth, and B. Schiele, "Pictorial structures revisited people detection and articulated pose estimation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 1014–1021.

[27] X. Tao and Z. Yun, "Fall prediction based on biomechanics equilibrium using Kinect," *International Journal of Distributed Sensor Networks*, vol. 13, no. 4, 2017.

[28] J. Tompson, R. Goroshin, A. Jain, Y. Lecun, and C. Bregler, "Efficient object localization using convolutional networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 648–656.

[29] 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," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4929–4937.

[30] "ECCV 2018 Joint COCO and Mapillary Recognition," [Accessed 18 April 2019]. [Online]. Available: http://cocodataset.org/#home

[31] "MSCOCO Keypoints Challenge 2017)," [Accessed 18 April 2019]. [Online]. Available: https://places-coco2017.github.io

[32] A. Toshev and C. Szegedy, "DeepPose: Human pose estimation via deep neural networks," in *IEEE Conf. Computer Vision and Pattern Recognition*, 2014, pp. 1653–1660.

[33] S.-e. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, "Convolutional pose machines," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4724–4732.

[34] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele, "2D human pose estimation: New benchmark and state of the art analysis," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2014, pp. 3686–3693.

[35] M. Andriluka, U. Iqbal, E. Insafutdinov, L. Pishchulin, and M. P. I. Informatics, "PoseTrack: A benchmark for human pose estimation and tracking," in *IEEE Conf. Computer Vision and Pattern Recognition*, 2018, pp. 5167–5176.

[36] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6299–6308.

[37] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask R-CNN," in *IEEE International Conference on Computer Vision*, 2017, pp. 2961–2969.

[38] D. Tome, C. Russell, and L. Agapito, "Lifting from the deep: Convolutional 3D pose estimation from a single image," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 5689–5698.

[39] H.-s. Fang, Y. Xu, W. Wang, X. Liu, and S.-c. Zhu, "Learning pose grammar to encode human body configuration for 3D pose estimation," in *AAAI Conference on Artificial Intelligence*, 2018.

[40] N. Sarafianos, B. Boteanu, B. Ionescu, and I. A. Kakadiaris, "3D human pose estimation : A review of the literature and analysis of covariates," *Computer Vision and Image Understanding*, vol. 152, no. 7, pp. 1–20, 2016.

[41] "How to use random sample consensus model," 2014. [Online]. Available: http://pointclouds.org/documentation/tutorials/random_sample_consensus.php

[42] "Opencv library," 2018, [Accessed 19 April 2019]. [Online]. Available: https://opencv.org/

[43] "Kinect for Windows SDK v1.8," 2012, [Accessed 18 April 2019]. [Online]. Available: https://www.microsoft.com/en-us/download/details.aspx?id=40278

[44] "International Research Institute MICA," 2019, [Accessed 19 April 2019]. [Online]. Available: http://mica.edu.vn/

[45] Z. X, "A study of microsoft Kinect calibration," George Mason University, Tech. Rep., 2012.

[46] J.-Y. B., "Camera calibration toolbox for Matlab," 2019, [Accessed 19 April 2019]. [Online]. Available: http://www.vision.caltech.edu/bouguetj/calib_doc/

[47] L. Sigal, A. O. Balan, and M. J. Black, "HumanEva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion," *International Journal of Computer Vision*, vol. 87, no. 1-2, p. 4, 2010.

[48] COCO, "Observations on the calculations of COCO metrics," 2019, [Accessed 24 April 2019]. [Online]. Available: https://github.com/cocodataset/cocoapi/issues/56

[49] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person pose estimation," [Accessed 23 April 2019]. [Online]. Available: https://github.com/ZheC/Realtime_Multi-Person_Pose_Estimation

[50] Tensorflow, "Tf-pose-estimation," [Accessed 23 April 2019]. [Online]. Available: https://github.com/ildoonet/tf-pose-estimation

[51] H. Wang, W. P. An, X. Wang, L. Fang, and J. Yuan, "Magnify-Net for multi-person 2D pose estimation," in *IEEE International Conference on Multimedia and Expo*, July 2018, pp. 1–6.

[52] caffe2, “Caffe2-pose-estimation,” [Accessed 23 April 2019]. [Online]. Available: https://github.com/eddieyi/caffe2-pose-estimation

[53] “Chainer realtime multi-person pose estimation,” [Accessed 23 April 2019]. [Online]. Available: https://github.com/DeNA/Chainer_Realtime_Multi-Person_Pose_Estimation

[54] mxnet, “Reimplementation of human keypoint detection in mxnet,” [Accessed 23 April 2019]. [Online]. Available: https://github.com/dragonfly90/mxnet_Realtime_Multi-Person_Pose_Estimation

[55] “MatConvNet realtime multi-person pose estimation,” [Accessed 23 April 2019]. [Online]. Available: https://github.com/coocoky/matconvnet_Realtime_Multi-Person_Pose_Estimation

[56] “CNTK realtime multi-person pose estimation,” [Accessed 23 April 2019]. [Online]. Available: https://github.com/Hzzone/CNTK_Realtime_Multi-Person_Pose_Estimation

[57] L. Bo and C. Sminchisescu, “Twin Gaussian processes for structured prediction,” *International Journal of Computer Vision*, vol. 87, no. 1-2, p. 28, 2010.

Nguyen Tuong Thanh received B.E. degree from Hanoi University Science and Technology in 2002 in Electronics and Telecommunications; He received M.E. degree in Electronic Engineering, University of Transport and Communications. He is now PhD student in Electronic Engineering, Hanoi University of Science and Technology. Currently, he is working at the Faculty of Engineering and Technology, Quy Nhon University. His research interests include computer vision, image processing 2-D, 3-D machine leaning, deep learning.

Le Van Hung received M.Sc. degree at Faculty Information Technology- Hanoi National University of Education (2013). He received PhD degree at International Research Institute MICA HUSTC-NRS/UMI - 2954 - INP Grenoble (2018). Currently, he is a lecture of Tan Trao University. His research interests include Computer vision, RANSAC and RANSAC variation and 3-D object detection, recognition, machine leaning, deep learning.

Pham Thanh Cong received M.Sc. degree at Electronics and Telecommunications, Hanoi University of Technology in 1998. He received PhD degree at Electronics and Telecommunications, Turin Polytechnic University, Italy in 2010. Currently, he is a lecturer of institute of Electronics and Telecommunications, Hanoi University of Science and Technology. His research interests include Super high frequency technology, antennas, telecommunication systems.