BULLETIN OF THE POLISH ACADEMY OF SCIENCES
TECHNICAL SCIENCES, Vol. 71(6), 2023, Article number: e147923
DOI: 10.24425/bpasts.2023.147923

An autonomous system for identifying and tracking characters using neural networks

Sebastian SŁOMIŃSKI and Magdalena SOBASZEK*

Warsaw University of Technology, Electrical Power Engineering Institute, Lighting Technology Division, Poland

Abstract. For the proper operation of intelligent lighting, the precise detection of a human silhouette on the scene is necessary. Correctly adjusting the light beam divergence requires locating the detected figure in virtual three-dimensional coordinates in real time. The market is currently dominated by the markers systems. This paper is focused on the advanced solution of the markerless system of identifying and tracking characters based on deep learning methods. Analyses of the selected pose detection, holistic detection (including BalzePose and MoveNet models), and body segmentation (BlazePose and tfbodypix) algorithms are presented. The BlazePose model was implemented for both pose tracking and body segmentation in the markerless dynamic lighting and mapping system. This article presents the results of the accuracy analysis of matching the displayed content to a moving silhouette. An assessment of the illumination precision was done as the function of the movement speed for the system with and without delay compensation.

Key words: markerless tracking; deep learning detection, dynamic lighting; pose identification.

1. INTRODUCTION

Dynamic lighting consists of creating a lighting scene according to the current position and pose of a moving object on the stage. Therefore, the lighting systems include devices that follow moving objects, as well as lighting devices such as moving heads and projectors. In more advanced systems, the devices used to estimate the position of the markers can be complemented by those that recognize real objects, people, etc. The advanced solutions used for dynamic object lighting are available on the market. Their advantage is the possibility of seeing the lighting effect using a built-in camera. However, the additional transmitter-receiver unit connected to the media server is necessary to track an object. FollowSpot, Robe BMFL FollowSPot, T1 Profile FS, Forte TS, etc. [1] are examples of such devices. It is essential to pay attention to the development of semi-automatic and partially autonomous products that still require the operator's control. The RoboSpot MotionCamera is an example of such a device. It looks like a moving head but is equipped with a camera. Thanks to this camera, it is possible to follow objects on the stage and synchronise their direction with the light from the lighting units.

In dynamic lighting systems, the precision of detecting and estimating the position of a moving object and its characteristic points in three-dimensional space is of key importance. This object tracking can be obtained in three ways: marker-based, markerless, and hybrid, combining both tracking variants. Most commercial solutions [2–4] for tracking moving characters are based on the use of markers. These types of systems require the use of passive or active markers. They are placed on the target moving object in the form of mini passive or active external devices or even special clothes. X, Y, and Z coordinates of the object in space and the information about its movement and rotation in three directions are determined by the position of the infrared transmitters or the information from radio transmitters. The prediction of characteristic points is limited to the position of the specific markers used. This is the significant limitation of these types of systems.

As far as markerless tracking is concerned, identifying the object and predicting its characteristic points are based on the analysis of images from scene digital recording devices, e.g., RGB, IR, and depth cameras [5–12]. This identification method is very complex and requires a lot of data throughput, but it offers more possibilities than marker-based tracking. It allows one to track any number of objects and also to estimate their characteristic points, such as human skeletal points, without additional devices. This identification method has the potential to estimate not only the position of the character itself in three-dimensional space but the information about its pose, size, or texture as well.

The precision of directing moving fixtures and matching the displayed graphic content to the moving object in real time are the key elements of dynamic lighting and mapping systems of moving objects. Moreover, it is crucial that in such systems the identification follows the dynamics of the illuminated object. Even the object identification should anticipate its position based on predicting the direction and calculating movement speed. Thus, the precision of the dynamic lighting is influenced by three aspects: the detection of the object in three-dimensional space, the accuracy of identification and the prediction of its characteristic points, and the delay. However, delays, shifts, or identification errors are unacceptable in the case of the dynamic mapping of objects with graphic content. So, computing and image processing time is critical. It is one of the

*e-mail: 01027471@pw.edu.pl

Manuscript submitted 2023-04-14, revised 2023-08-11, initially accepted for publication 2023-10-15, published in December 2023.

reasons why such advanced, commercially-ready systems are essentially non-existent.

The problem of pose estimation in images is widely studied in the field where machine learning (ML) methods are used. It should be noted that over recent years deep learning (DL) models have been researched extensively. This research focuses on the solutions for estimating the poses based on RGB images from one or more cameras simultaneously. The use of DL methods has resulted in the development of algorithms with increased efficiency and precision in pose estimation.

1.1. Aim and scope of the research

The purpose of this research is to identify and implement a reliable, markerless system for identifying positions and body poses with the use of deep learning algorithms. This research aims to eliminate the limitations of systems based on tracking motion with the use of markers (most often infrared or radio radiators). The implementation of a markerless system of active tracking of the human silhouette in the scene requires fast processing of images from recording devices in real time. In addition, the key aspect of markerless pose tracking is the high efficiency and precision of identifying the pose in motion for each recorded frame of the video material. The research presented in this article aims to demonstrate the possibility of using deep learning methods in the original, markerless, dynamic lighting system and projection mapping of a moving silhouette. The scope of this research covers two main problems. The first one concerns the analysis of the selected human detection algorithms in the identification accuracy and the throughput of the analysed input data. The second one relates to the implementation of selected DL algorithms in the target system which enables, among others, real-time illumination and mapping with the graphic content of moving characters. The algorithms for intelligent identification of the human figure and motion prediction are implemented to compensate for the mapped object speed and direction of movement in real time.

2. LITERATURE REVIEW

By analysing existing solutions based on Deep Learning in the context of detecting human silhouette parts, the current models can be divided into categories depending on the obtained output information. The first group of models consists of the models that detect a specific part of the body, e.g., a head, a face [13–17], eyes, hands, or the whole body. The second group includes the models for single-pose detection [7, 18] or multiple-pose detection [6, 9, 19, 20]. In the case of face detection, a significant difference between available approaches concerns the number of the estimated characteristic points, including eyes, a mouth, a nose, or a face oval. An additional difference is whether the returned values apply only to the coordinates of the 2D image or whether they are three-dimensional coordinates with the information about the depth. The depth is usually calculated considering one selected point, e.g., in the case of detected face points, the centre of the head [17] is the reference point. When estimating the pose of the whole human body, the midpoint between the hips [7] is the reference.

The approaches to face or head detection involve models that estimate the area with a detected face in the form of the so-called bounding box (Fig. 1a) [21, 22]. Other models also return information about five (Fig. 1b) [15] or six [14] characteristic facial points, i.e., eyes, nose, and mouth corners. The third group concerns the models that estimate 68 points of the face (Fig. 1c) [16, 23, 24] which include its contours, eyes, mouth, nose, and eyebrows. The received points are in the form of 2D or 3D points. The most advanced models facilitate the estimation of 468 2D or 3D facial points (Fig. 1d) [17]. If these are 3D points, it is possible to reproduce the whole face geometry.

In the process of human pose detection and estimation through image analysis, convolutional neural networks (CNN) with two different implementation approaches are mainly used. These approaches are bottom-up [6, 9] and top-down [20, 25, 26]. In the first solution, all limbs are detected first, and then the poses are estimated. In the top-down approach, the areas with people are predicted then the human pose in each region is calculated.

The detection and prediction of skeletal points of the whole body are realised by the models for single pose detection [7, 18] or multiple pose detection [6, 9, 20, 26, 27]. Additionally, these models differ in the number of the estimated skeletal points. The MPII model (Fig. 2a) is the simplest model and it facilitates the estimation of 15 human body points. The COCO model (Fig. 2b) is another example that predicts 17 points of the body. Compared with the previously mentioned models, the BlazePose model (Fig. 2c) facilitates the calculation of as many as 33 points.

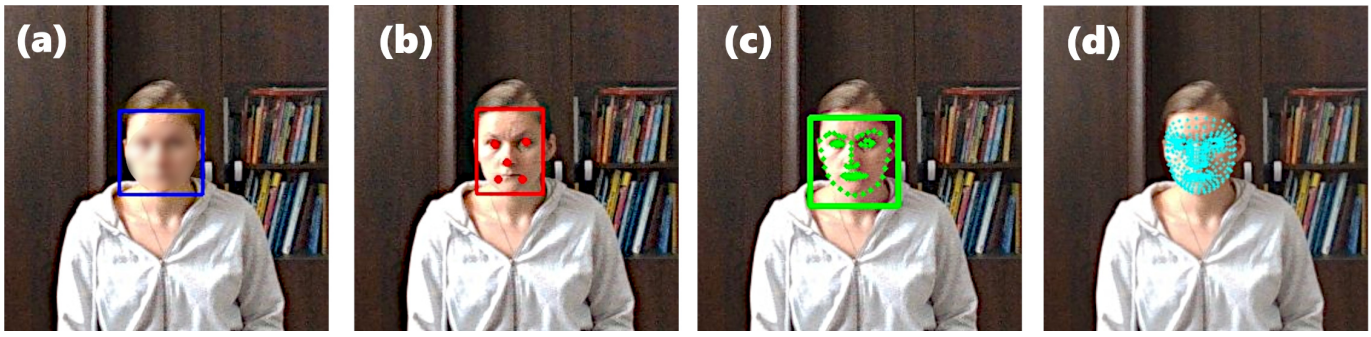

Fig. 1. Face detection models as (a) a bounding box with a face, (b) a bounding box with a face and its five points, (c) a bounding box and 68 facial points (d) 468 facial points

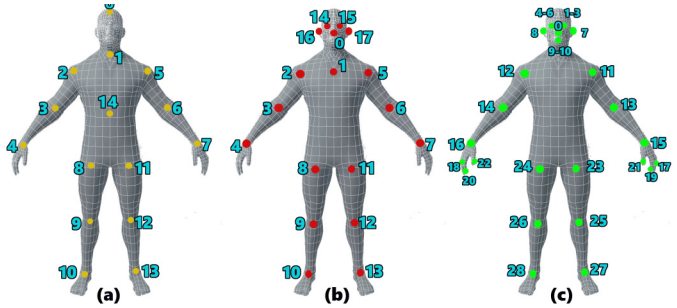

Fig. 2. Pose estimation models (a) the MPII model, (b) the COCO model, and (c) the BlazePose model

The BlazePose model [7] is a convolutional neural network that allows 33 skeletal points of one pose to be estimated as 3D coordinates (Fig. 2c). The depth reference point is the midpoint between the hips. In addition to the information about the coordinates in the image and Z, the model returns a visibility parameter for each estimated point. It indicates whether the point is visible from the camera point of view. The model operation is based on two steps. Firstly, the ROI (region of interest) in which the pose is located is predicted and its resolution of the network input image size is adjusted, i.e., to 256×256. Then, the pose is estimated as 33 skeleton points. Moreover, the body segmentation is possible. A mask is predicted in which pixels not belonging to the character take the value 0 and the others take the value 1. The full model runs at 10 fps on the GPU of a Pixel 2 mobile device [7].

MoveNet [18] is an example of a model estimating 17 2D body points (Fig. 2b). The model is available in two versions that differ in speed and precision analysis, and they are referred to as a lightning model and a thunder model. The first model accepts the image with a 192×192 px resolution as an input, whereas a thunder version is intended for a 256×256 px image size. MoveNet is the bottom-up model that was trained on the Active (Google's internal set) and COCO datasets. MobileNetV2 [28] is the feature extractor in the MoveNet model.

In addition to the solutions for pose estimation and face detection, some models allow for the obtaining of information such as body segmentation and body parts segmentation [7, 19]. Body segmentation (Fig. 3a) involves the extraction of pixels from an image representing the entire body, while body part segmentation (Fig. 3b) allows specific image pixels to be assigned to particular body parts.

BodyPix [19, 27] is an example of a body segmentation approach. BodyPix is an open-source model using convolutional neural networks, enabling body segmentation, body part segmentation, and 18-point 2D detection of multiple poses from an image. For this approach, Resnet50 [29] and MobileNetV1 [30] were used as base networks. In addition, it is possible to adjust the accuracy and tracking speed using three parameters: outputStride (8, 16, or 32), multiplier (1, 0.75, or 0.5), and quantBytes (control of the bytes used to quantise the weight which can take values of 1, 2 or 4).

In the scientific literature, there are approaches for 3D pose detection and estimation systems based on the analysis of RGB images from multiple cameras. This type of CNN system is presented in [31–36]. The authors in [32] propose a 3D pose estimation approach based on images from unsynchronised and uncalibrated cameras. In [31], the prediction of 2D points in each view was made using HR-Net [37]. The 3D pose reconstruction was based on the plane sweep stereo. The authors of [31] used the Cascade Pyramid Network [38] to estimate the 2D poses on the corresponding camera images. A multi-view alignment algorithm was used to identify the poses in the images and the 3D poses were estimated on the basis of this algorithm. The paper [36] presents the estimation of the 3D pose by the flexible selection of the number of recording cameras and IMUs placed on the tracked person. OpenPose was used to detect 2D poses on each image [6].

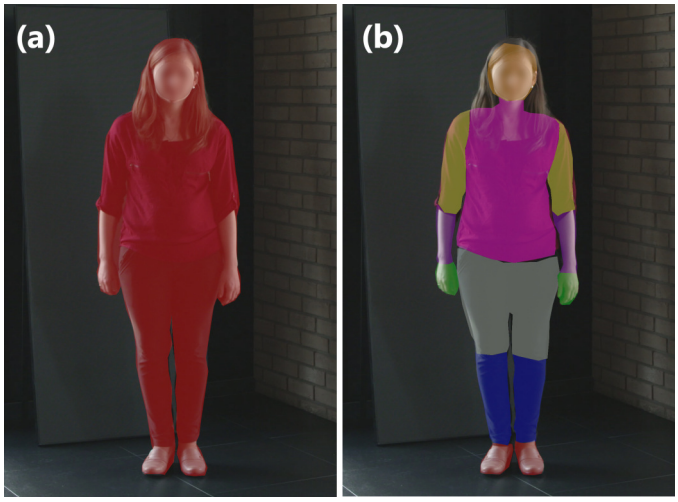

Fig. 3. (a) Body segmentation and (b) body part segmentation

It is worth highlighting that DL methods offer many possibilities for tracking moving figures on the stage, such as the prediction of skeletal points, detection of specific body parts and their characteristic points, or a body segmentation from an image. That is why using the DL method will enable the implementation of dynamic lighting in a markerless way.

3. RESEARCH SYSTEM DESCRIPTION

The research [8, 39] shows the possibility of realising a dynamic lighting system and mapping with any content of moving objects based on markerless identification. The system facilitates the illumination of specific parts of the human body with the possibility of adjusting the light spot and its colour [39]. In addition, equipping the system with projectors as illumination units allows the surfaces of moving objects, e.g. things [8] or characters [39] to be mapped precisely without going beyond their contours in real time. Another advantage is the possibility of reducing the discomfort glare phenomenon by using a multimedia projector system for lighting purposes. This can be achieved by lowering the luminance levels for selected body parts such as the face or eye zones [39]. It is possible to eliminate the lighting pollution [40] of the natural environment by masking objects from the surroundings for projectors used as

a device to illuminate the objects. Furthermore, this approach provides the opportunity to dynamically dark people who can enter the area of the illuminating beam and experience a discomfort glare.

In the research [8,39], background subtraction and corner detection of the tracked objects were implemented to identify flat or cuboidal objects. Background subtraction and the information about the detected poses returned by the RGB-D Kinect v2 camera were used to track a moving character [39].

In the authors' research, in order to eliminate the limitations of Kinect-type devices and the need for precise identification, it was decided to analyse the available machine learning (ML) methods for detecting and tracking moving figures. In particular, it was decided to focus on the methods based on deep neural networks. The objective of the performed analysis was to investigate the possibility of implementing these object detection methods in markerless lighting systems. The analysis consisted of determining the delay level resulting from the operation of a given character identification algorithm as the function of the resolution of the input images. Furthermore, the efficiency of the correct detection of the selected algorithms was investigated.

It was decided to choose the models that enable tracking the entire silhouette: BlazePose model and two MoveNet models for single pose detection. In addition, the algorithms for body segmentation were investigated. The approach available in the MediaPipe library is the first of these algorithms. BodyPix is the second model and it allows one to segment both the whole body and the specific body parts. Additionally, the performance of the algorithm for holistic tracking based on the BalzePose and BlazeFace models (also available in the MediaPipe library) was conducted. This solution returns 468 face points, 25 hand points each, and 33 human skeletal points.

Following the analysis of the image processing speed and detection efficiency of the selected algorithms, a markerless real-time dynamic lighting and mapping system was performed. In this system, the BlazePose model and the body segmentation available in the MediaPipe library were used to detect and track a moving character on the stage. In addition, proprietary algorithms were implemented to compensate for the latency caused by the total analysis, processing time of the images, and the dynamics of the character's movement. The description of the system and its calibration process are presented in Section 3.2, and the results of the system without and with delay compensation are presented in Section 4.4.

3.1. Description of ML approaches implementation

The machine learning algorithms that enable pose detection (Fig. 2), holistic detection, and body segmentation (Fig. 3) were analysed. The performed tests consisted of analysing the data throughput as the function of the resolution of the input image for each tested detection algorithm. All algorithms were tested on the same data set to ensure the reliability of the obtained results. Own video material was used for the analysis: the sequence of 348 images for pose detection, holistic detection, and body segmentation. The tests were performed for six input image resolutions. These resolutions were 320×240 px, 640×480 px, 1280×960 px, 1920×1440 px, 2560×1920 px, and 3840×2880 px. These image sizes were used to analyse the BlazePose and BodyPix models. However, 192×192 px and 256×256 px image sizes were considered for the MoveNet models.

The mentioned algorithms were implemented and tested using the Python 3.9.7 programming language in the Windows 10 operating system. The implementation of a pose detection models consisted of:

- BlazePose – a pose detection – using the functions available in the MediaPipe library for two model variants differing in computational complexity and identification precision.
- MoveNet – a pose detection – the lighting and thunder models were implemented using the TensorFlow library.
- BlazePose + BlazePalm – a holistic detection – MediaPipe library was also used for the implementation of this approach.
- BodyPix – a body segmentation – one fastest MobileNetV1 model was implemented using the tfbodypix python package.
- BlazePose – a body segmentation – MediaPipe python package was used.

The speed analysis of the DL algorithms was carried out on the unit consisting of an Intel Core i7-9750H 6-Core Processor 2.4 GHz (4.5 GHz) 16 GB RAM with an NVIDIA GeForce GTX 1660 Ti 8 GB mobile graphics card. The benchmark tests were carried out in order to determine the performance of this hardware configuration. Three kinds of benchmark software were used: Novabench ("NovaBench," n.d.), CineBench R15, and CineBench R20 ("Cinebench," n.d.). Cinebench is a tool based on the Cinema 4D engine, designed to test the processor and graphic card. NovaBench evaluates the general performance of a computer by assessing the processor, graphic card, RAM, and disk. In the NovaBench test, the used PC scored a total of 2838 pts, including 1294 pts CPU, 1104 pts GPU, 273 pts RAM, and 167 pts for disk read and write speeds. For the Cinebench tests, the CPU reached the values of 3416 pts and 1020 pts in the R20 and R15 tests, respectively. In the Cinebench R15 test, the GPU obtained a rendering speed of 139 fps.

3.2. The description of the dynamic lighting system and its calibration

In order for the dynamic lighting system to work correctly, a calibration of the camera-projector setup is first required. It consists of determining the transformation matrix from the camera to the projector layout (Fig. 4a). This transformation is defined by the following formula (1):

$$\begin{bmatrix} X_p \\ Y_p \\ Z_p \\ 1 \end{bmatrix} = \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix}. \tag{1}$$

Then taking into account the conversion of 3D points to a 2D plane for the pinhole camera model, the following dependence

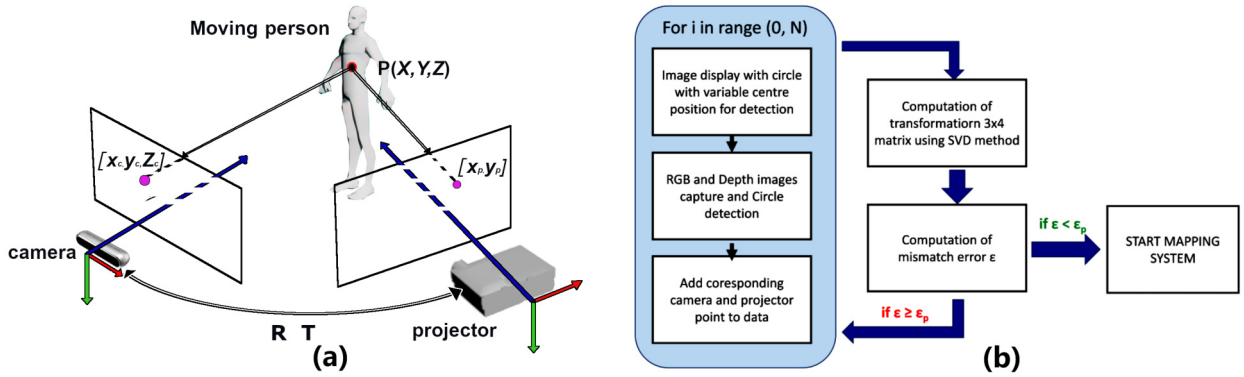

Fig. 4. (a) The point transformation from the camera to the projector layout, system. (b) diagram of the calibration process of the camera-projector

is obtained:

$$\begin{bmatrix} w \cdot x_p \\ w \cdot y_p \\ w \end{bmatrix} = \begin{bmatrix} q_1 & q_2 & q_3 & q_4 \\ q_5 & q_6 & q_7 & q_8 \\ q_9 & q_{10} & q_{11} & q_{12} \end{bmatrix} \begin{bmatrix} x_c Z_c \\ y_c Z_c \\ Z_c \\ 1 \end{bmatrix}. \qquad (2)$$

It means that each pixel of the image displayed by the projector can be calculated based on the images from the RGBD camera. For this purpose, a 3×4 calibration matrix has to be determined. Based on the 2D points of the camera image and their depth information from the depth map this matrix enables the 2D points of the projector image to be calculated. The transformation matrix can be obtained from the corresponding camera points (x_c, y_c, Z_c) and projector points (x_p, y_p) and solving the obtained system of linear equations.

The process of calibration is shown in Fig. 4b. The display resolution is automatically read out, and the image of the projector native resolution is created in order to enable calibration flexibility. First, the camera and projector points correlating with each other are determined. For this, sequential images are displayed with a single circle with variable but known coordinates of its centre (Fig. 5a). For each of the images, a displayed circle is detected, and the coordinates of its centre are read out in the RGB image of the camera (Fig. 5b). At the same time, the depth values are read in the depth map (Fig. 5c). The circle detection must be carried out for different values of the distance between the plane (which is used to display the image) and the projector. Next, the system of linear equations is created from the corresponding coordinates. The 3×4 transformation matrix is calculated using the SVD method (formula (2)). Then the mismatch error is calculated to verify the calibration of the camera-projector system. This error is determined as the largest difference between the projector image points which were used to calculate the calibration matrix and the points estimated with the obtained calibration matrix and the camera points. In case the obtained error is too high, the registration of points is repeated, and the calibration matrix and mismatch error are recalculated. For example, for a 1920×1080 px projector resolution, a maximum error value of 3 px was considered acceptable.

The dynamic lighting and mapping system for the moving figure consists of the projector, the RealSense D455 RGBD camera, and a computer (Fig. 6a). The projector and camera are positioned to provide the largest possible common field of view. In the analysed system configuration, the projector and camera covered a field of view of 2.55×1.5 m and 5×2.9 m, respectively. The camera is set to capture an RGB image and depth map of 1280×720 px resolution with 30 fps, which is the highest available depth image size. The captured RGB image is matched to the viewpoint of the depth map. The system enables the lighting of the moving character in real time. Additionally, it is possible to display any graphic content on the silhouette using a projector. By detecting the parameters and the pose of the human figure, it is possible to adjust the image to the body surface precisely.

The mapping process is shown in Fig. 6b. First, the RGB image and depth map are registered simultaneously. Next, pose detection and body segmentation are realised using the BlazePose

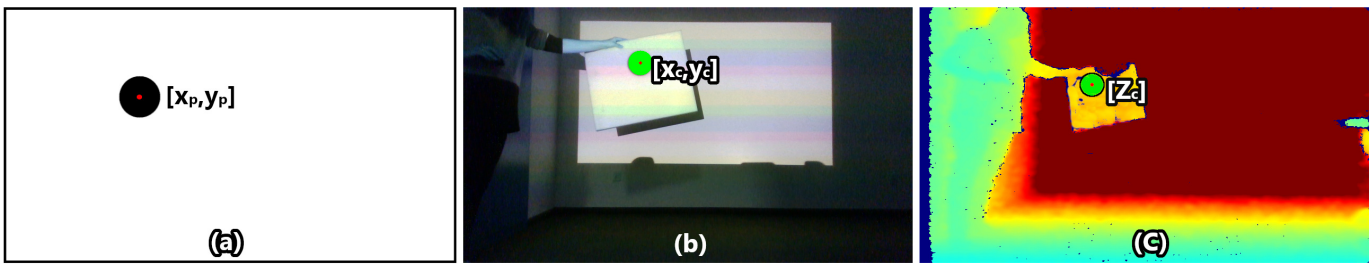

Fig. 5. The calibration process a) the image with one circle displayed by a projector, b) circle detection in the RGB image, c) reading the corresponding depth value of the circle centre on the depth map

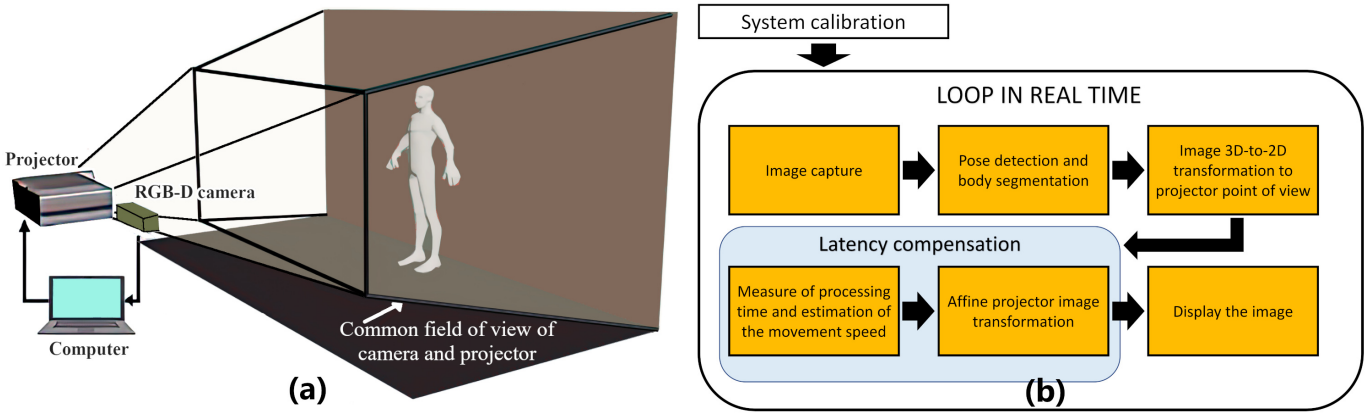

Fig. 6. (a) Dynamic lighting system, (b) diagram of the dynamic lighting process of a moving figure

model. Pose tracking consists of the detection of 33 3D skeletal points. Then, a 3D-to-2D transformation is applied to the image with the binary character mask obtained in the previous step. This image transformation adjusts the image to the resolution and field of view of the projector based on the calibration matrix (formula (2)). Next, this image is converted to an RGB image and multiplied with the selected image or colour in order to add colour or texture. In order to increase the precision of matching the displayed content to the moving figure surface, the system facilitates compensation for the delay caused by image detection and transformation. A block diagram of the delay and offset of the displayed graphic content compensation process is shown in Fig. 7.

The latency compensation is based on estimating the speed of movement based on the current and previous frames. The compensation consists of calculating the shift between the projected image and the moving figure position in the three directions X, Y, and Z. The estimation of these three offsets is done using the coordinates of the head midpoint in the previous $[x_{pi}-1,\ y_{pi}-1,\ Z_{ci}-1]$ and current frame $[x_{pi},\ y_{pi},\ Z_{ci}]$ from the projector point of view. Additionally, the time between the registration of two consecutive frames and the figure tracking and image processing time tp in the current frame is considered. The mismatch for the current frame is calculated from the following formula:

$$\begin{aligned} \Delta X &= (x_{pi}-x_{pi-1})\cdot\frac{t_p}{t_o}, \qquad \Delta Y = (y_{pi}-y_{pi-1})\cdot\frac{t_p}{t_o}, \\ \Delta Z &= (Z_{ci}-Z_{ci-1})\cdot\frac{t_p}{t_o}, \qquad \text{scale} = 1-\frac{\Delta Z}{Z_{ci}}. \end{aligned} \tag{3}$$

The X- and Y-axis offset is calculated in pixels, while the Z-axis displacement is calculated in mm. In order to eliminate delay compensation errors, the final values of the ΔX, ΔY and ΔZ shifts are determined as the arithmetic means of the last ten calculated displacements for each considered direction with the rejection of outliers. Next, these data are used to estimate the affine transformation matrix applied to the display image. This transformation takes into account the X- and Y-direction image shifts and the scaling of the image based on the Z-axis offset in the following formula (4):

$$\begin{bmatrix} x'_p \\ y'_p \\ 1 \end{bmatrix} = \begin{bmatrix} \text{scale} & 0 & \Delta X \\ 0 & \text{scale} & \Delta Y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_p \\ y_p \\ 1 \end{bmatrix}. \tag{4}$$

4. RESULTS ANALYSIS

4.1. Pose detection processing time

Two approaches were analysed to detect and estimate key body points of a pose: MoveNet and BlazePose. For the first solution, two models were tested: Lightning and Thunder. They differ in the size of the input image (192×192 px and 256×256 px) which affects the precision of identification. These models were obtained using the TensorFlow Lite to ensure the highest possible performance on the CPU. For MoveNet, the analysis was conducted for CPU and GPU. In the case of the CPU, the analysis times were 9.81 ± 2.5 ms for Lightning and 33.22 ± 3.4 ms for Thunder model. For GPU, these models reached the times of 12.17 ± 4.5 ms and 13.74 ± 5 ms, respectively. It can be seen that for the less precise model, the obtained difference in times

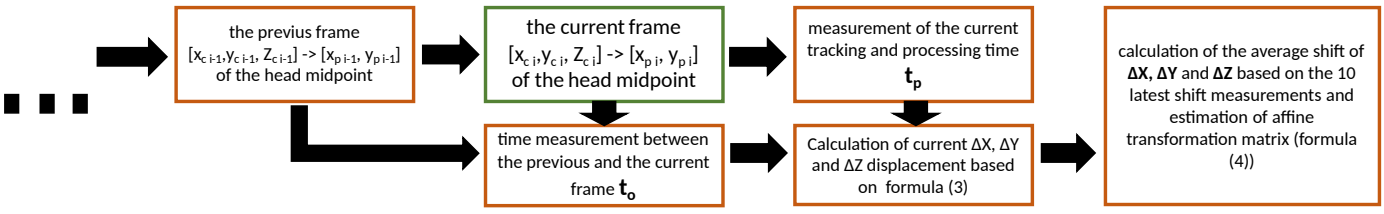

Fig. 7. Diagram of the delay compensation

between CPU and GPU is about 2.5 ms. For the Thunder model, the use of GPU as a computational unit reduced the detection time by 20 ms in comparison to the CPU.

For BlazePose, a CPU performance analysis was carried out for two complexity factors equal to 0 and 1. A higher value indicates the model with higher estimation precision of body points. The analysis of the processing data speed was performed as a function of the input image resolution. The obtained results for the used CPU are shown in Fig. 8. For a 320×240 px image resolution, the processing times were around 16.0 ± 1.1 ms for a model complexity of 0 and 23.5 ± 2 ms for a model with a complexity of 1. For the highest considered resolution, the computational time of these models increased to 22.6 ± 1.9 ms and 30.4 ± 2 ms.

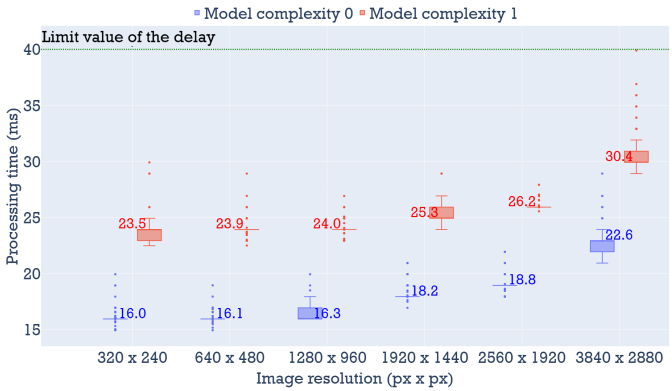

Fig. 8. Processing time for the two BlazePose models depending on the resolution of the input image

The results obtained for MoveNet and BlazePose solutions are below 40 ms. So, these pose-tracking methods enable the realisation of dynamic lighting and video mapping in real time.

4.2. Body segmentation and holistic detection processing time

In addition to face detection or character body point prediction, the models based on neural networks enable complex solutions such as holistic detection. Furthermore, there are CNN-based methods for body segmentation or body parts segmentation from an image. Therefore, the data processing analysis was conducted for two solutions available in the MediaPipe library – body segmentation and holistic detection. Additionally, the models for the body and its parts segmentation available in the tfbodypix library (corresponding to the model for pose tracking) were tested. The analysis was carried out on CPU and the results are shown in Fig. 9.

Analysing the results for holistic detection, the processing times obtained for CPU were in the range of 33.1 to 40.8 ms (Fig. 9a). For MediaPipe body segmentation, these times were 3.2 ms to analyse the 320×240 px image resolution and 7.2 ms to analyse the largest image resolution.

BodyPix body segmentation approach was slower than MediaPipe segmentation (Fig. 9b). For a 320×240 px resolution, the analysis time was twice as long as the result obtained for the BlazePose model. For a 640×480 px frame size, the average processing time was 22.3 ms. Whereas the analysis time increased to over 600ms for the highest image resolution tested. In the case of a body part segmentation, the shortest time was 25.1 ms for 320×240 px resolution. For higher resolutions, the processing time was longer than 40 ms.

4.3. The detection efficiency analysis

The analysis was conducted for the BlazePose and MoveNet solutions in order to verify the efficiency and correctness of silhouette detection. Two image sequences depicting the fully visible moving figure which motion included movement around a circle. The images included the figure sideways (Fig. 10b), forwards (Fig. 10(a, c)), and backwards (Fig. 10d) in relation to the camera position. In addition, the videos differed in the recording lighting conditions. The first sequence of images was recorded in a darkened room with little artificial lighting (at a level of 50 lx) (Fig. 10(c,d)). The second sequence of images was recorded in a lobby where daylight was the main light source (Fig. 10(a,b)). In addition, the first room was small and the second room was fully equipped and furnished.

The analysis of detection efficiency was conducted depending on the resolution of the input image. The considered image sizes were from 320×180 px to 3840×2160 px for BlazePose,

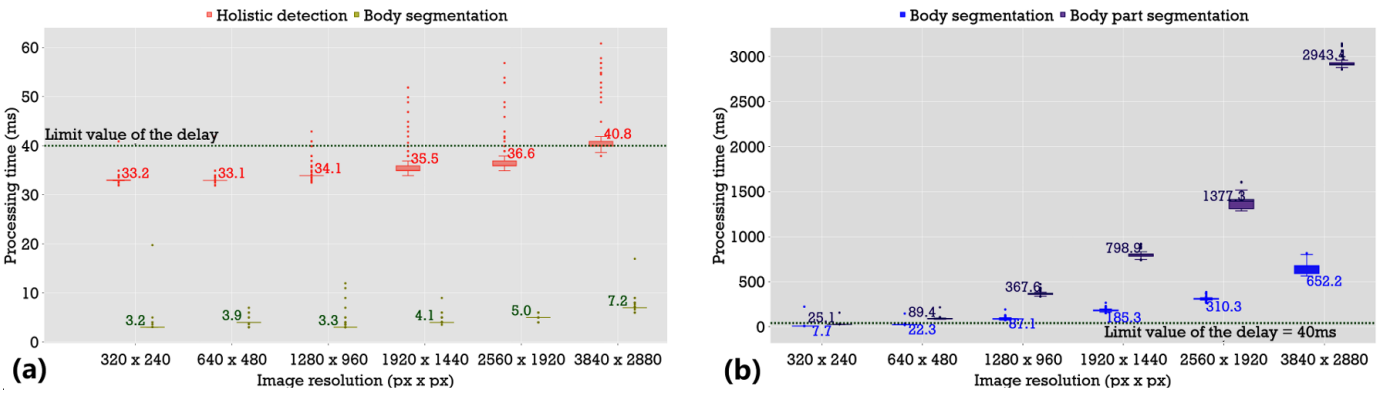

Fig. 9. Processing time for a) holistic detection and body segmentation from MediaPipe library, b) body and body part segmentation available in tfbodypix python package

Fig. 10. Examples of images used to analyse the effectiveness of the algorithms (a) pose viewed from the front, (b) view of the pose from side in the lobby and (c) view of the pose from the front, (d) view of the pose from the backside in the darkening room

192 × 192 px and 256 × 256 px for MoveNet models. The efficiency analysis assessed the ratio of the number of frames with correct detection to all analysed frames. Additionally, the average number of estimated pose points was examined.

Table 1 shows the results of the percentage effectiveness of pose detection algorithms for two different lighting conditions and eight input image resolutions. The average number of estimated points for pose detection was 33 points for Blaze Pose and 17 points for MoveNet Thunder. These results are equal to the number of points returned by a given pose detection solution. In addition, analysing the results obtained for the pose detection solutions using BlazePose and MoveNet, it can be seen that the detection efficiency was 100% for all considered cases. These results are shown in Table 1. This means that in all processed images, the pose was detected correctly. Nevertheless, it should be noted that the prediction precision of the skeleton points varied depending on the considered room and the size of the input image.

Table 1
Analysis of the efficiency of pose detection depending on the resolution of the input

Resolution	MoveNet		Resolution	BlazePose	
[px × px]	Room 1	Room 2	[px × px]	Room 1	Room 2
192 × 192	100%	100%	320 × 180	100%	100%
256 × 256	100%	100%	640 × 360	100%	100%
			960 × 540	100%	100%
			1920 × 1080	100%	100%
			2880 × 1620	100%	100%
			3840 × 2160	100%	100%

The higher precision of a pose estimation was obtained by the BlazePose model than by the MoveNet model. Figure 11 shows the difference in the precision of point estimation in the head and torso regions. It is because in the MoveNet Thunder solution the processed image is scaled to the resolution of the input image dedicated to this model, i.e. 256 × 256 px. Then, detection and pose estimation take place for the scaled image. On the other hand, the BlazePose solution first detects the area with the figure in the image without scaling. This image region is then matched to the resolution of the model, i.e. 256 × 256 px.

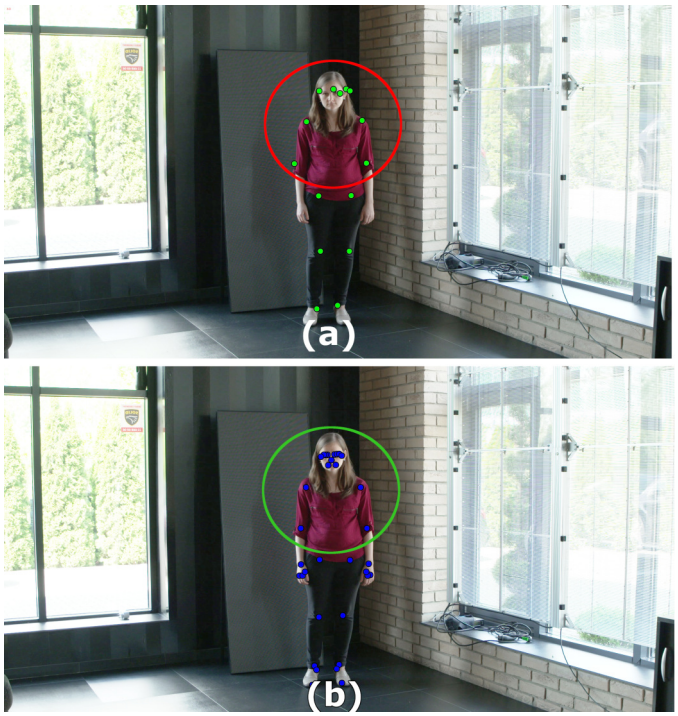

Fig. 11. Comparison of pose detection precision by (a) the MoveNet Thunder model with (b) the BlazePose model

4.4. Dynamic lighting and the delay compensation test

The analysis of the accuracy of the dynamic illumination of a moving figure was performed for three different speeds of its movement slow, normal, and fast. The dynamic illumination system described in Section 3.2 was used for the tests. The figure movement included the movement to the right, left side, and backwards to the camera. The average processing time per frame was 37.7 ms, while the average time between two consecutive shots was 53.5 ms.

The additional striped pattern was projected using a second projector with a resolution of 1024 × 768 px (Fig. 12a) to determine the movement speed and analyse the shifts resulting from the latency. The width of one strip was 26px which translated into the width of 6.8 cm. The effect of a dynamic illumination of the moving figure for the considered speeds was recorded with a high-speed acA1920 Basler 156 fps camera.

The calculation of the velocity of the figure movement and the estimation of mismatch errors were performed on the basis of the recorded image sequences for the three-figure motions. The error analysis assessed the offset between the figure and

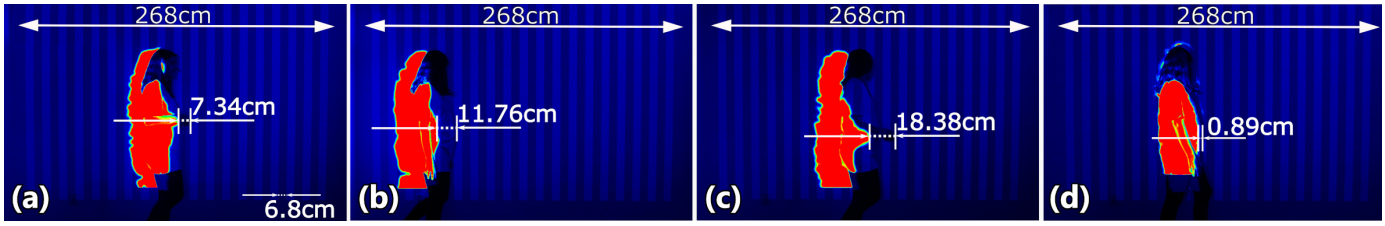

Fig. 12. (a) Mismatch error of Motion I, (b) mismatch error of Motion II, (c) mismatch compensation for Motion II error of Motion III, (d) adjustment result for delay

the displayed image frame by frame (Fig. 12). The velocity was estimated based on the total displacement of the body and the calculated time of this movement based on the number of over-analysed frames.

The alignment analysis of the displayed content to the moving person was done for the system without and with delay compensation. The delay compensation is based on estimating and limiting displacements in three directions: X, Y, and Z between the displayed image and the moving person (Fig. 7 and formula (3), (4)). Table 2 shows the mismatch error results for the three tested figure gait speeds. For the variant without delay compensation, the estimated character movement speeds were 57.6 cm/s, 88.9 cm/s, and 122.1 cm/s. The average mismatch error was 7.34 cm (Fig. 12a), 11.76 cm (Fig. 12b), and 18.38 cm (Fig. 12c), considered from the lowest to the highest velocity. The employment of delay compensation significantly reduced the average mismatch error of the displayed content to the moving figure (Fig. 12d). The average error was 0.78 cm for slow motion, 0.89 cm for normal motion, and 1.18 cm for fast motion.

Table 2

Analysis of the mismatch error of the dynamic illumination system of a moving character without and with delay compensation for three motion speeds of the figure

	MOTION I – SLOW		MOTION II – NORMAL		MOTION III – FAST	
	V [cm/s]	error [cm]	V [cm/s]	error [cm]	V [cm/s]	error [cm]
Without delay compensation	57.6	7.34	88.9	11.76	122.1	18.38
Delay compensation	51.7	0.78	75.2	0.89	123.5	1.18

5. CONCLUSIONS

This paper presents the results of the study that have a significant impact on the realisation of dynamic lighting and object mapping. The system is based on pose estimation and body segmentation in a markerless system using deep learning (DL) methods. It should be remembered that the successful implementation of dynamic lighting requires two conditions. The first one is identifying the object position and its parts correctly. The second condition is to calculate the speed and predict the direction of the movement of the object in order to adjust the displayed content to its surface properly. It should be noted that resolution matching is a crucial element of the optimisation of the analysed images. This determines the processing time to obtain smooth video mapping and lighting effects.

The key part of the paper includes the analysis of the processing time of the selected algorithms for pose detection and body segmentation as a function of the resolution of the input data. Additionally, the detection efficiency of the analysed algorithms and the implementation of BlazePose model in the dynamic lighting system of a moving figure were examined. Finally, the results of the system without and with the delay compensation are presented.

For a single pose detection, the average processing time for both analysed solutions, MoveNet and BlazePose, is below 40 ms. In the case of body segmentation, the time of less than 10 ms for each resolution analysed was achieved by the solution available in the MediaPipe library (Fig. 9a). The models available in the tfbodypix library have significantly lower performance. The results of 22.3 ms and 25.1 ms were obtained for the 640×480 px and 320×240 px resolutions for body and body part segmentation, respectively (Fig. 9b).

For the research to be complete, the detection efficiency analysis was done. It is essential to realise that comparing processing times does not directly correlate to a positive detection effect. The pose detection efficiency of BlazePose and MoveNet models was analysed as the function of image resolution. The experiment was conducted to demonstrate how precision is influenced by changing the resolution of the processed image to make calculations faster. In the case of pose detection algorithms, the detection efficiency was 100% for each analysed resolution (Table 1). However, it needs to be emphasised that the higher precision of the skeleton point estimation is obtained by the BlazePose model (Fig. 11). The lower precision of the MoveNet models is due to the limited resolution of the input image to be only 256×256 px.

The obtained results regarding the data throughput and the high pose detection performance of the DL methods demonstrate the feasibility of implementing these methods in a real-time dynamic illumination system for moving objects. The effect of such a system is presented by the results in Table 2 and Fig. 12. The results obtained for a system without delay compensation show that object tracking and image processing times result in a large mismatch in the displayed content to the moving person (from 7.34 to 18.38 cm). Thus, it is necessary to

implement delay-compensating algorithms in this type of system. These algorithms make it possible to significantly increase the precision of illumination and mapping of moving figures in real time. For the three analysed pose movement speeds, the delay compensation reduces the mismatch error from as much as 18.4 cm to around 1 cm (Fig. 12).

The results of experiments and analysis prove that it is possible to conduct online image analysis for pose detection for dynamic video mapping using machine learning algorithms. Thanks to that, it is possible to eliminate problematic radio and infrared markers used in the existing dynamic lighting systems.

ACKNOWLEDGEMENTS

This work was supported by internal grants in 2021 to support the conduct of scientific activity in the discipline of Automatic Control, Electronics and Electrical Engineering at the Warsaw University of Technology.

REFERENCES

[1] "Robe Lighting," https://www.robelighting.com/ (accessed May 04, 2022).

[2] "Blacktrax," https://blacktrax.cast-soft.com/ (accessed May 04, 2022).

[3] "OptiTrack," Available: https://www.optitrack.com/, (accessed May 04, 2022).

[4] "Xsens," https://www.xsens.com/ (accessed May 04, 2022).

[5] A.M. Ghonim, W.M. Salama, A.A.M. Khalaf, and M.H. Shalaby, "Indoor localization based on visible light communication and machine learning algorithms," *Opto-Electron. Rev.*, vol. 30, p. 140858, 2022, doi: 10.24425/opelre.2022.140858.

[6] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, "OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, pp. 172–186, Dec. 2021, doi: 10.1109/TPAMI.2019.2929257.

[7] V. Bazarevsky, I. Grishchenko, K. Raveendran, T. Zhu, F. Zhang, and M. Grundmann, "BlazePose: On-device Real-time Body Pose tracking," *ArXiv*, Jun. 2020, [Online]. Available: http://arxiv.org/abs/2006.10204.

[8] S. Słomiński and M. Sobaszek, "Intelligent object shape and position identification for needs of dynamic luminance shaping in object floodlighting and projection mapping," *Energies (Basel)*, vol. 13, no. 23, p. 6442, Dec. 2020, doi: 10.3390/en13236442.

[9] S. Kreiss, L. Bertoni, and A. Alahi, "OpenPifPaf: Composite Fields for Semantic Keypoint Detection and Spatio-Temporal Association," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 8, pp. 13498–13511, Aug. 2022, doi: 10.1109/TITS.2021.3124981.

[10] J. Cheng, L. Zhang, Q. Chen, and R. Long, "Position detection for electric vehicle DWCS using VI-SLAM method," *Energy Rep.*, vol. 7, pp. 1–9, Nov. 2021, doi: 10.1016/j.egyr.2021.09.086.

[11] K. Mohammed, A.S. Tolba, and M. Elmogy, "Multimodal student attendance management system (MSAMS)," *Ain Shams Engineering Journal*, vol. 9, no. 4, pp. 2917–2929, Dec. 2018, doi: 10.1016/j.asej.2018.08.002.

[12] N. Aunsri and S. Rattarom, "Novel eye-based features for head pose-free gaze estimation with web camera: New model and low-cost device," *Ain Shams Eng. J.*, vol. 13, no. 5, p. 101731, Sep. 2022, doi: 10.1016/j.asej.2022.101731.

[13] S. Sharma, K. Shanmugasundaram, and S.K. Ramasamy, "FAREC – CNN based efficient face recognition technique using Dlib," in *Proceedings of 2016 International Conference on Advanced Communication Control and Computing Technologies, ICACCCT 2016*, IEEE, Jan. 2017, pp. 192–195, doi: 10.1109/ICACCCT.2016.7831628.

[14] V. Bazarevsky, Y. Kartynnik, A. Vakunov, K. Raveendran, and M. Grundmann, "BlazeFace: Sub-millisecond Neural Face Detection on Mobile GPUs," *ArXiv*, Jul. 2019, [Online]. Available: http://arxiv.org/abs/1907.05047.

[15] K. Zhang, Z. Zhang, and Qiao Yu, "Joint Face Detection and Alignment using Multi-task Cascaded Convolutional Networks," *IEEE Signal Process. Lett.*, vol. 23, pp. 1499–1053, 2016, doi: 10.1109/LSP.2016.2603342.

[16] A. Bulat and G. Tzimiropoulos, "How far are we from solving the 2D & 3D Face Alignment problem? (and a dataset of 230,000 3D facial landmarks)," *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, doi: 10.1109/ICCV.2017.116.

[17] Y. Kartynnik, A. Ablavatski, I. Grishchenko, and M. Grundmann, "Real-time Facial Surface Geometry from Monocular Video on Mobile GPUs," *ArXiv*, Jul. 2019, [Online]. Available: http://arxiv.org/abs/1907.06724.

[18] "MoveNet," https://www.tensorflow.org/hub/tutorials/movenet (accessed May 04, 2022).

[19] A. Mankotia and M. Meenu Garg, "Real-time person segmentation," *Int. J. Creat. Res. Thoughts (IJCRT)*, vol. 9, no. 6, pp. 30–36, 2021, [Online]. Available: https://www.ijcrt.org/papers/IJCRT2106125.pdf.

[20] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," *2017 IEEE International Conference on Computer Vision (ICCV)*, Mar. 2017, doi: 10.1109/ICCV.2017.322.

[21] P. Viola and M. Jones, "Rapid Object Detection using a Boosted Cascade of Simple Features," *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, pp. 1–9, 2001, doi: 10.1109/CVPR.2001.990517.

[22] J. Chmielińska and J. Jakubowski, "Detection of driver fatigue symptoms using transfer learning," *Bull. Pol. Acad. Sci. Tech. Sci.*, vol. 66, no. 6, pp. 869–874, 2018, doi: 10.24425/bpas.2018.125934.

[23] S. Suwarno and K. Kevin, "Analysis of Face Recognition Algorithm: Dlib and OpenCV," *J. Inform. Telecomm. Eng.*, vol. 4, no. 1, pp. 173–184, Jul. 2020, doi: 10.31289/jite.v4i1.3865.

[24] G. Anbarjafari, R.E. Haamer, I. LÜSi, T. Tikk, and L. Valgma, "3D face reconstruction with region based best fit blending using mobile phone for virtual reality based social media," *Bull. Pol. Acad. Sci. Tech. Sci.*, vol. 67, no. 1, pp. 125–132, 2019, doi: 10.24425/bpas.2019.127341.

[25] H.-S. Fang, S. Xie, Y.-W. Tai, and C. Lu, "RMPE: Regional Multi-person Pose Estimation," *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 2353–2362, Nov. 2017, doi: 10.1109/ICCV.2017.256.

[26] J. Li, C. Wang, H. Zhu, Y. Mao, H.-S. Fang, and C. Lu, "CrowdPose: Efficient Crowded Scenes Pose Estimation and A New Benchmark," *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Dec. 2019, doi: 10.1109/CVPR.2019.01112.

[27] "BodyPix," https://blog.tensorflow.org/2019/11/updated-bodypix-2.html (accessed May 04, 2022).

[28] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jan. 2018, doi: 10.1109/CVPR.2018.00474.

[29] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Dec. 2015, doi: 10.1109/cvpr.2016.90.

[30] A.G. Howard *et al.*, "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," *ArXiv*, Apr. 2017, [Online]. Available: http://arxiv.org/abs/1704.04861.

[31] J. Lin and G.H. Lee, "Multi-View Multi-Person 3D Pose Estimation with Plane Sweep Stereo," *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, doi: 10.1109/CVPR46437.2021.01171.

[32] K. Takahashi, D. Mikami, M. Isogawa, and H. Kimata, "Human Pose as Calibration Pattern; 3D Human Pose Estimation with Multiple Unsynchronized and Uncalibrated Cameras," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 1888–1895, 2018, doi: 10.1109/CVPRW.2018.00230.

[33] J. Dong, W. Jiang, Q. Huang, H. Bao, and X. Zhou, "Fast and Robust Multi-Person 3D Pose Estimation from Multiple Views," *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–10, Jan. 2019, doi: 10.1109/CVPR.2019.00798.

[34] C. Huang *et al.*, "End-to-end Dynamic Matching Network for Multi-view Multi-person 3d Pose Estimation," *European Conference on Computer Vision*, pp. 477–493, 2020, doi: 10.1007/978-3-030-58604-1_29.

[35] H. Chen, P. Guo, P. Li, G.H. Lee, and G. Chirikjian, "Multi-person 3D Pose Estimation in Crowded Scenes Based on Multi-View Geometry-Supplementary Material," *Lecture Notes in Computer Science*, pp. 541–557, 2020, doi: 10.1007/978-3-030-58580-8_32.

[36] C. Malleson, J. Collomosse, and A. Hilton, "Real-Time Multi-person Motion Capture from Multi-view Video and IMUs," *Int. J. Comput. Vis.*, vol. 128, no. 6, pp. 1594–1611, Jun. 2020, doi: 10.1007/s11263-019-01270-5.

[37] K. Sun, B. Xiao, D. Liu, and J. Wang, "Deep High-Resolution Representation Learning for Human Pose Estimation," *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, doi: 10.1109/CVPR.2019.00584.

[38] Y. Chen, Z. Wang, Y. Peng, Z. Zhang, and G. Yu Jian Sun, "Cascaded Pyramid Network for Multi-Person Pose Estimation," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7103–7112, 2018, doi: 10.1109/CVPR.2018.00742.

[39] S. Słomiński and M. Sobaszek, "Dynamic autonomous identification and intelligent lighting of moving objects with discomfort glare limitation," *Energies (Basel)*, vol. 14, no. 21, p. 7243, Nov. 2021, doi: 10.3390/en14217243.

[40] K. Skarżyński and W. Żagan, "Improving the quantitative features of architectural lighting at the design stage using the modified design algorithm," *Energy Rep.*, vol. 8, pp. 10582–10593, Nov. 2022, doi: 10.1016/j.egyr.2022.08.203.