Integrated In-vehicle Monitoring System Using 3D Human Pose Estimation and Seat Belt Segmentation

Ginam Kim [†], [1] **Hyunsung Kim** [†], [1] **Joseph Kihoon Kim,** [1] **Sung-Sik Cho,** [2]
Yeong-Hun Park, [2] **Suk-Ju Kang,** [1]. [*]

[1] Sogang University
[2] Hyundai Mobis
{ginamkim, ghkskxlr, deepwork, sjkang}@sogang.ac.kr
{sscho, yhpark0119}@mobis.co.kr

Abstract

Recently, along with interest in autonomous vehicles, the importance of monitoring systems for both drivers and passengers inside vehicles has been increasing. This paper proposes a novel in-vehicle monitoring system the combines 3D pose estimation, seat-belt segmentation, and seat-belt status classification networks. Our system outputs various information necessary for monitoring by accurately considering the data characteristics of the in-vehicle environment. Specifically, the proposed 3D pose estimation directly estimates the absolute coordinates of keypoints for a driver and passengers, and the proposed seat-belt segmentation is implemented by applying a structure based on the feature pyramid. In addition, we propose a classification task to distinguish between normal and abnormal states of wearing a seat belt using results that combine 3D pose estimation with seat-belt segmentation. These tasks can be learned simultaneously and operate in real-time. Our method was evaluated on a private dataset we newly created and annotated. The experimental results show that our method has significantly high performance that can be applied directly to real in-vehicle monitoring systems.

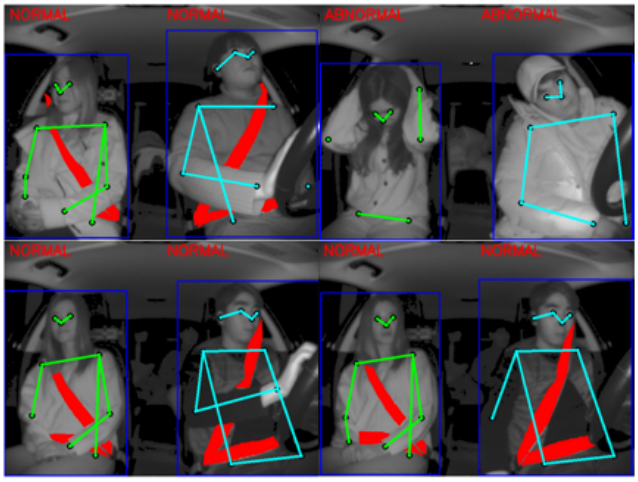

Figure 1: Overview of our dataset and proposed network.

1 Introduction

Convolutional Neural Networks (CNNs) are widely applied to advanced driver assistance systems for autonomous driving (**????**). These systems are generally used to process various information gathered from outside vehicles such as outside object detection and line segmentation. However, monitoring the conditions, behaviors, and seat-belt-wearing status of a driver and their passengers is very important to reduce the risk of accidents. In particular, the classification accuracy between normal and abnormal states of wearing a seat belt might help prevent fatalities or serious injury. However, existing in-vehicle monitoring systems have limitations in terms of classifying the condition, behavior, and seat-belt status of the driver and passengers. The CNNs in an in-vehicle monitoring system (IVMS) can simply solve these problems using a vision sensor. This paper proposes 3D human pose estimation to identify the conditions and behaviors of a driver and passengers and proposes a novel classification network for normal/abnormal seat-belt wearing. The results of our network can be adopted to give an alarm to passengers to improve safety.

Our architecture consists of the three following modules. First, we propose the absolute keypoints coordinate estimation method based on conventional 2D/3D human pose estimation networks (**??**). Second, the proposed seat-belt segmentation network consists of parallel deconvolution structures. Third, the seat-belt wearing status classification is performed using the results of those two above mentioned networks and high-resolution features from the backbone network. The entire network is trained in an end-to-end manner, and it shows remarkable performance.

Generally, in-vehicle monitoring systems require an infrared (IR) camera to operate robustly regardless of the luminance change, unlike typical 3D human pose estimation. This means that a new dataset is necessary for in-vehicle monitoring because the IR dataset has different characteristics from typical RGB images. Additionally, since the 3D human pose dataset is generally produced under multi-view points, producing a new 3D human pose dataset has a high annotation cost. We solved this problem using the in-vehicle environment characteristics. Since previous datasets (**??**) contain various positions of human objects, each image has a variety of root-depth. Therefore, rather than directly esti-

[*]† The first two authors are equally contributed for this paper.

mate the absolute depth of each keypoint, our method estimates the relative depth of each keypoint with an additional network that estimates the depth of the root keypoint. However, the variation of depth values in vehicles is limited. Furthermore, in most situations inside vehicles, this has almost a fixed value, unlike situations outside the vehicle. Therefore, each keypoint can be directly estimated without an additional root depth-estimating network.

In these characteristics of in-vehicle monitoring, we annotate 2D keypoints using an infrared (IR) camera and depth values with a Time of Flight (ToF) camera. We use the depth value of the ToF camera as the ground truth depth. As a result, a 3D human pose dataset is produced with only a 2D keypoints annotation cost, thereby significantly reducing the annotation cost. Our private dataset includes consists of a total of 30,000 images. The contributions of this paper are summarized as follows.

- We propose a novel end-to-end network that integrates 3D human pose estimation, seat-belt segmentation, and seat-belt status classification. To our knowledge, 3D human pose estimation was first applied inside a vehicle.
- A new insight for a data generation method is proposed to consider the characteristics of the vehicle's internal environment.
- Our proposed method shows remarkable performance that can be directly applied to a real in-vehicle monitoring system that operates in real-time.

2 Related Works

3D Human Pose Estimation 3D human pose estimation is mainly categorized into top-down and bottom-up methods. Top-down methods use a cropped bounding box as input that contains a single person (**?????????**). Meanwhile, bottom-up methods estimate all personal keypoints from the input image and then group them into each set of a person (**?????**). Our proposed method taken the form of a top-down method.

Top-down methods have two conventional approaches. One is the single-stage approach that directly estimates the 3D coordinates from an input cropped image (**?????**). (**?**) trained regression and detection tasks simultaneously to obtain an accurate 3D human pose. (**?**) adopted structure-aware regression, which showed that the regression-based method is more effective than the detection-based method for pose estimation. The network of (**?**) estimated the 3D human pose in a coarse-to-fine manner by applying CNNs iteratively. Therefore, the CNN refined the image features at every step. (**?**) proposed the soft-argmax operation to tackle issues caused by post-processing and quantization errors; this can be applied to any heatmap-based 3D pose estimation network to obtain coordinates with its differentiable property. (**?**) adopted the conventional idea of 2D multi-person top-down approaches to 3D multi-person pose estimation; they detected human bounding boxes using a human detector and then estimated the keypoints for each person. (**?**) used RootNet to estimate the absolute root location and PoseNet to estimate the root-relative coordinates; their method showed significant improvement in terms of 3D multi-person pose estimation.

The alternative is a two-stage approach with a lifting network (**????**). The two-stage method first estimates 2D keypoints' coordinates and then translates 2D coordinates into 3D coordinates using an additional lifting network. (**?**) proposed a simple and lightweight lifting network that could predict 3D human poses with given 2D keypoint locations. (**?**) adopted Long Short-Term Memory (LSTM) to predict the depth of keypoints. The two types of LSTM used the results of 2D pose estimation and input image patches as input; they achieved better performance lifting 2D keypoints to 3D keypoints. (**?**) used attention-based transformer encoder blocks to predict 3D keypoints; the inputs for this method were a sequence of 2D keypoints and the network generated 3D keypoints. (**?**) proposed an online augmentation method that could generate harder poses to estimate. Using the harder cases, the entire 3D pose estimation network learned various geometry factors of human poses.

Those two approaches generally estimate the depth value of the root keypoint and then the depth of each keypoint to add this to the root depth to produce the final output. Their adoption of this method lies in the characteristics of the dataset. The commonly used datasets (**??**) have various depths of human objects in images. Some people exist nearby, and those who are far away also exist at the same time. Since the network cannot effectively estimate the wide depth range of the data, one keypoint (pelvis) is set as the root keypoint, the depth value of which is extracted by a separately designed network. Therefore, the keypoints estimation network estimates only the relative depth at each keypoint. This method showed effective performance.

Human pose estimation for in-vehicle monitoring system Recently developed 2D/3D human pose estimation networks using deep learning have shown remarkable performance. However, pose estimation networks for IVMS have not improved much. Only a few networks (**????**) have attempted to assess the performance in an in-vehicle environment, and even those have focused solely on 2D pose estimation. (**?**) proposed an architecture that estimated human pose and face orientation for an autonomous driving system that consisted of only three convolutional layers and a fully connected layer; through this shallow network, it can perform real-time processing. (**?**) suggested predicting only the arms of the driver and passengers; this network used partial affinity fields (PAF) from (**?**). (**?**) has the most similar architecture to our proposed network; they performed 2D pose estimation and seat-belt segmentation and used PAF to estimate 2D keypoints, but they only estimated body keypoints without face keypoints.

Seat belts Efforts have been made to solve seat-belt-aware tasks such as detection, segmentation, and status classification in the area of computer vision, but trials to apply CNN remain in their infancy. (**?**) tried to detect seat-belt by edge detection using a salient gradient. (**?**) performed seat-belt status classification using Tiny-YOLO (**?**). First, they detected the main part of the seat-belt and corner using Tiny-YOLO and then classified whether the seat-belt was fastened correctly. (**?**) (as mentioned above) performed seat-belt segmentation using a feature pyramid network during simultaneous 2D human pose estimation.

Clothes		
Jacket, long-sleeve	short-sleeve	winter clothes
34%	33%	33%

Table 1: Subject statistics.

3 Proposed Methods

Our goal is to detect absolute 3D semantic keypoint coordinates of the driver and front passenger in a top-down manner and perform seat-belt segmentation using a single ToF camera. Finally, in this paper, our proposed network performs a seat-belt status classification. Figure 4 describes the overall architecture of the proposed method, which is composed of 3D pose estimation, seat-belt segmentation, and seat-belt classification. In the absolute 3D pose estimation, we extract the heatmaps of keypoints using the conventional CNN architecture. For the accurate seat-belt segmentation masks, we adopt the deconvolution layer-based parallel architecture to all output features in the backbone network and then use the output of those networks and the high-resolution feature as input. The following sections describe this in more detail.

3.1 Dataset generation

The biggest bottleneck to applying CNN-based computer vision tasks in IVMS is appropriate training datasets. Few datasets are tailored to in-vehicle environments, so we manufactured a dataset to train our proposed network including 30K images. Moreover, we propose an efficient methodology to manufacture this dataset for the in-vehicle environment with relatively low cost. We set up IR and ToF cameras inside a vehicle to collect data on the driver and passengers. The ToF camera can collct and robustly operate depth information regardless of luminance changes. As summarized in Table 1, the driver and passengers changed clothes several types to consider the situation of various seasons for almost 20 people. Each outfit accounts for 33% of the total dataset. During data collection, we assumed various scenarios that may occur while driving. These scenarios include various general actions such as getting on and off, adjusting the seat position, operating an infotainment system, and operating a handle, as well as other actions such as stretching, and wearing a hat or a coat.

Our 3D absolute keypoints estimation network estimates the absolute 3D coordinates of keypoints from images cropped by detecting the human. In this case, the depth values for the driver and passengers in the vehicle are almost unchanged. Figure 2 shows that most keypoints are distributed 400–900 mm. In particular, 64.26% of the keypoints exist within 500–800 mm and 96.83% are present within 400–900mm. This means that the depth value variation is limited in the environment inside a vehicle and the process of estimating the root depth value using an additional root-depth estimation network is unnecessary. From this observation, we can predict the absolute coordinates without any additional root-depth estimation networks.

In addition, as shown in Figure 3(b), the image from the IR camera should be normalized for use as input. In the

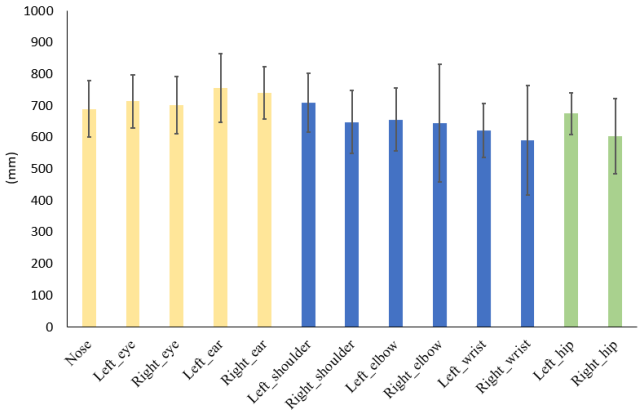

Figure 2: Depth value distribution of the keypoints

Figure 3: Collected image examples. (a) Normalized gray-scale image from the ToF camera. (b) Normalized gray-scale image from the IR camera.

annotation process, we used the format of the MSCOCO dataset (**?**), which is one of the most widely used datasets in object detection, semantic segmentation, and human pose estimation. Using this dataset, we first, made a bounding box for each person; thus, our dataset has only one object class (person). Second, 2D keypoint coordinates were annotated. The MSCOCO has 17 keypoints for every human, whereas, our dataset has only 13 points. In the in-vehicle environment, lower body parts are usually unseen; therefore we only collected the keypoints of the face and upper body. With the annotated 2D coordinates, we extracted the depth values at the same locations in the ToF raw data of Figure 3(a). Finally, we annotated the segmentation mask in the form of a polygon and divided the seat-belt status classes into normal or abnormal according to whether it is worn normally. A ToF camera was used for data generation. At the inference time, only the IR camera is used.

3.2 Proposed Network Architecture

3D absolute keypoints estimation. As described in Figure 4, the 3D absolute keypoints estimation network is composed of a backbone network and three deconvolution layers. It is a simple architecture that is widely used in 2D/3D keypoints estimation (**??**). We used ResNet50 (**?**) as a backbone network. The extracted backbone feature $\mathbf{F}_B$ becomes 3D keypoint heatmaps $\{\mathbf{H}_k\}_{k=1}^{K}$ after passing through the 4×4 deconvolution layer three times as follows:

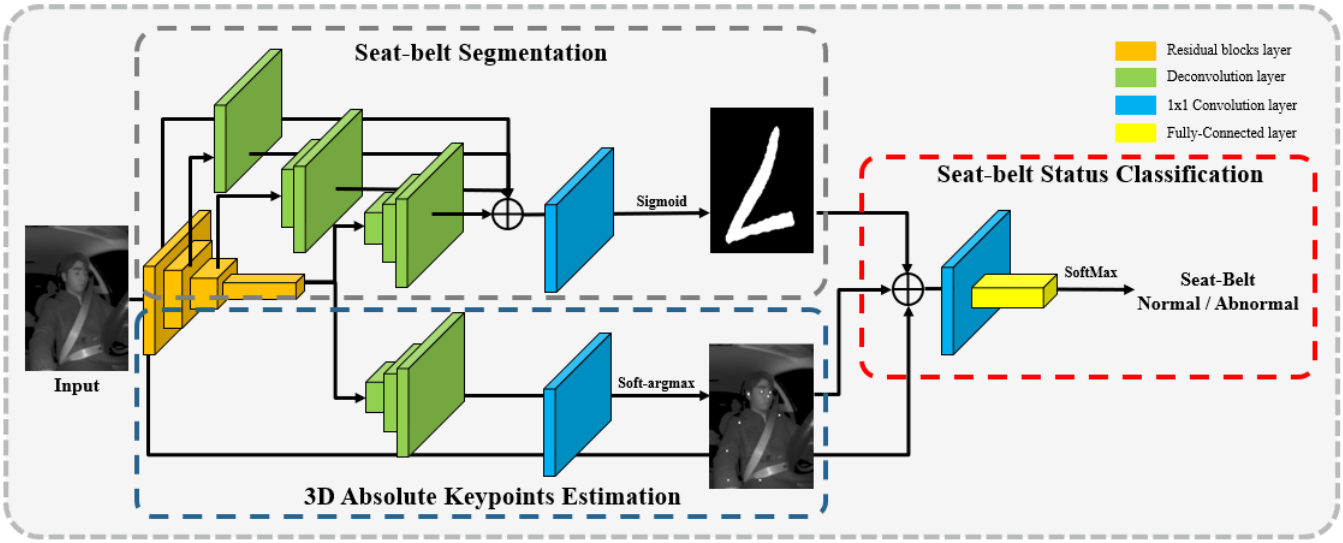

Figure 4: The overall architecture of the proposed system integrating 3D pose estimation, seat-belt segmentation, and seat-belt status classification.

$$\{\mathbf{H}_k\}_{k=1}^{K} = Deconv_{4\times4}^{3}(\mathbf{F}_{B4}), \quad (1)$$

where K is the total number of keypoints. Since obtaining the coordinate of maximum value is a non-differentiable operation, the soft-argmax operation is used to obtain the 3D keypoint coordinates $\{[x, y, z]_k\}_{k=1}^{K}$ from $\{\mathbf{H}_k\}_{k=1}^{K}$ as follows:

$$\{[x, y, z]_k\}_{k=1}^{K} = Soft\_argmax(\{\mathbf{H}_k\}_{k=1}^{K}). \quad (2)$$

Seat-belt segmentation. Seat-belt segmentation predicts a binary segmentation mask. The binary mask from the ground truth polygons is used as a ground truth segmentation. To predict an accurate segmentation mask, the segmentation network has a parallel deconvolution layer structure that applies deconvolution to 2nd, 3rd, and 4th layer outputs of the backbone, respectively, and all features that are estimated as the backbone deepens can be used. Each deconvolution layer has the same kernel size of 4×4 and is applied differently depending on the resolution until reaching the same size as the output of the 1st layer. The upsampled features are concatenated with the output of the 1st layer and pass through the convolution layer once more and become $F_{seatbelt}$. Finally, the sigmoid function is used to extract the binary seat-belt segmentation mask $Mask_seatbelt$.

$$Mask_{seatbelt} = Sigmoid(F_{seatbelt}), \quad (3)$$

Seat-belt status classification. The seat-belt status classification network uses $\{\mathbf{H}_k\}_{k=1}^{K}$, $F_{seatbelt}$ and the high-resolution feature F_H that comes from the first layer of the backbone as an input. Because $\{\mathbf{H}_k\}_{k=1}^{K}$ contains only heatmap information and $F_{seatbelt}$ describes the seat-belt segmentation mask, F_H is necessary to classify the seat-belt wearing status. Those features pass through the 1×1 convolution layer and a fully connected layer after being concatenated. Finally, with the softmax operation, the seat-belt status confidence score Cf_{status} is generated.

3.3 Loss function

We define the loss function for each task. The loss of 3D absolute keypoints estimation $L_{keypoints}$ is the Mean Absolute Error (MAE) which is calculated with $\{\mathbf{H}_k\}_{k=1}^{K}$ and the ground truth heatmap $\{\mathbf{H}gt_k\}_{k=1}^{K}$. Moreover the seat-belt segmentation loss L_{seg} and classification loss L_{cls} are Mean Squared Error (MSE), respectively as follows:

$$L_{keypoints} = \frac{1}{n} \times \sum \left| \{\mathbf{H}_k\}_{k=1}^{K} - \{\mathbf{H}gt_k\}_{k=1}^{K} \right|, \quad (4)$$

$$L_{seg} = \frac{1}{n} \times \sum \left| Mask_{seatbelt} - Mask_{gt} \right|^2, \quad (5)$$

$$L_{cls} = \frac{1}{n} \times \sum \left| Cf_{status} - Cf_{gt} \right|^2, \quad (6)$$

where n is the total size of the data, $Mask_{gt}$ means the ground truth seat-belt segmentation mask and Cf_{gt} is the ground truth one-hot vector of seat-belt status classes. The total amount of loss is calculated as follows:

$$L_{total} = L_{keypoints} + \alpha L_{seg} + L_{cls}, \quad (7)$$

where α is a hyper-parameter for L_{seg}. Using this loss function (7), our entire proposed network can be trained in an end-to-end manner.

4 Experiments

4.1 Implementation details

The proposed dataset contains $60,000$ person instances within $30,000$ images. We used 80% of the generated

	Keypoints		MPJPE(mm)	Distribution(%)		
				< 30mm	< 50mm	< 70mm
Driver	Face	nose	17.74	90.43	97.76	99.47
		eye	15.92	88.22	98.73	99.76
		ear	19.73	84.19	97.23	98.76
	Upper body	shoulder	33.96	53.93	85.35	96.02
		elbow	33.74	55.71	88.14	95.43
		wrist	56.83	50.29	71.88	80.46
	Lower body	pelvis	53.82	36.08	70.97	88.68
	Total		31.14 mm			
Passenger	Face	nose	33.95	58.76	79.55	91.64
		eye	56.13	41.13	64.05	77.93
		ear	60.01	22.80	49.76	74.47
	Upper body	shoulder	51.73	26.49	58.18	78.61
		elbow	54.72	31.09	60.16	77.67
		wrist	53.49	32.47	60.05	76.89
	Lower body	pelvis	57.34	23.71	55.30	74.15
	Total		19.49 mm			
Driver & Passenger	Face	nose	25.16	75.26	88.29	95.89
		eye	33.57	67.55	83.51	90.17
		ear	24.88	76.34	91.16	95.82
	Upper body	shoulder	41.68	42.00	73.54	88.45
		elbow	46.40	40.86	71.26	84.71
		wrist	54.77	39.31	64.58	78.26
	Lower body	pelvis	55.24	31.09	64.64	82.81
	Total		41.01 mm			

Table 2: 3D keypoints performance analysis on our dataset.

Driver					
Left Keypoints MPJPE (mm)			Right Keypoints MPJPE (mm)		
Face	Upper body	Pelvis	Face	Upper body	Pelvis
18.49	61.71	112.32	16.54	31.29	37.66
Passenger					
Left Keypoints MPJPE (mm)			Right Keypoints MPJPE (mm)		
Face	Upper body	Pelvis	Face	Upper body	Pelvis
57.06	47.68	38.97	55.77	62.05	75.80

Table 3: Comparison of the left and right body 3D keypoints MPJPE according to the driver and the passenger.

dataset as a training set, and the other 20% as a validation set. Our model was trained on the proposed training set without any extra data and experimental results were demonstrated on the validation set. The entire training and testing was performed with an NVIDIA GeForce RTX 3090 GPU. For the evaluation, the Mean Per Joint Position Error (MPJPE) is used as a 3D keypoints evaluation metric and the Interaction over Union (IoU) is employed as an evaluation metric for seat-belt segmentation. We used the Adam optimizer (**?**) and the models were initialized randomly. In the training phase, the initial learning rate was set to 1e−3, and dropped to 1e−4 at the 50th and 1e−5 at the 70th epochs, respectively. ResNet50 (**?**) was used as the backbone networks. We set α to 100.

	3D pose estimation	Seat-belt segmentation	Seat-belt classification	Total
Accuracy	41.01 mm (MPJPE)	80.64 % (IoU)	95.90 % (Accuracy)	-
Speed (FPS)	145.07	686.54	5824.67	129.03

Table 4: Entire network performance evaluation.

4.2 Results

We analyzed the 3D pose estimation results as summarized in Table 2; the results for the driver and front passenger were analyzed separately. When comparing the average values, the driver's MPJPE is 31.14mm, which is relatively lower than that of the passenger 52.26mm. Since we assumed actual driving situations when manufacturing the dataset, the driver concentrated on driving conditions and the passenger performed more malicious actions. The results for each keypoint show that overall, most keypoints were estimated to have an MPJPE within 70mm, and both the driver and passenger showed a lower MPJPE for the face keypoints than the upper body keypoints. In Table 3, a remarkable point is that the driver has a higher error in the left keypoints of their body than in the right, while the passenger shows the opposite. From these results, we can analyze that estimating the outside keypoints of both people is more complicated because outside keypoints are more vulnerable to occlusion due to the camera's angle of view and several objects. The MPJPE for the entire test set is 41.01 mm; it shows better

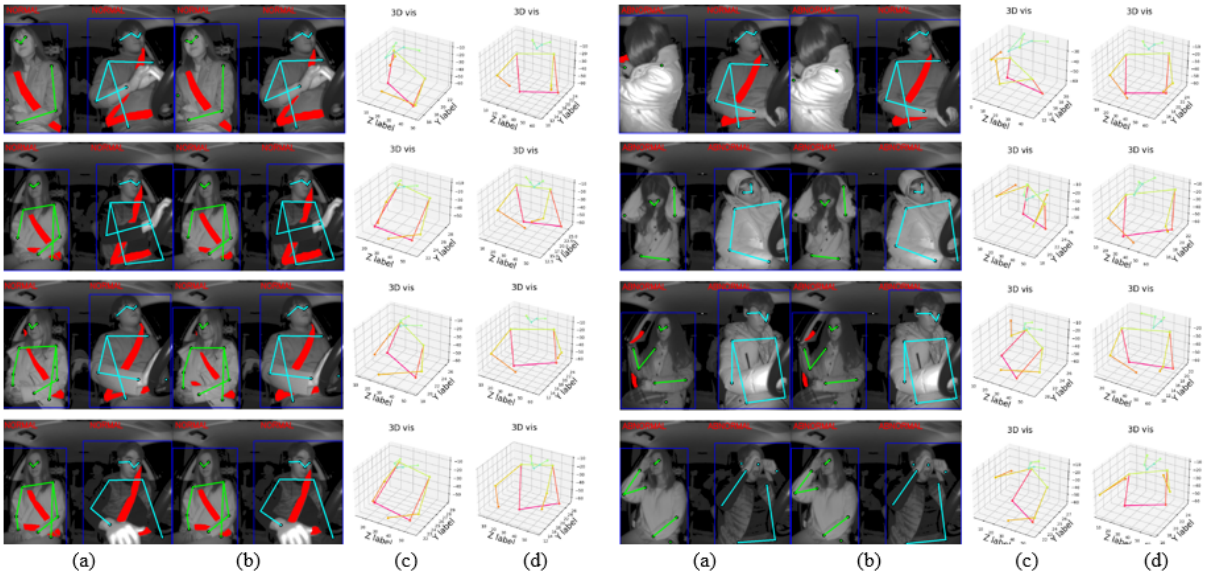

Figure 5: Estimated 3D human pose and seat-belt segmentation sample results. (a) ground truth, (b) estimated results, (c) 3D human pose estimation results of the passenger in the 3D domain, and (d) 3D human pose estimation results of the driver in 3D the domain.

performance than state-of-the-art networks have achieved in public datasets. These results prove that our proposed network is sufficiently effective to be applied directly in in-vehicle environments. As summarized in Table 4, we evaluated the overall network performance. As mentioned above, the 3D pose estimation performance shows an MPJPE of 41.01 mm, and the 3D pose estimation network operates at 145.07 fps. Seat-belt segmentation also has a high IoU performance of 80.64% and 686.54 fps in a single operation. Finally, the seat-belt classification shows high accuracy of 95.90%. The operation speed of the entire network is 129.03 fps using an NVIDIA 3090 RTX. As described in Figure 5, the qualitative results of our proposed network show remarkable performance. Our method implements seat-belt segmentation precisely even when little of the seat-belt is visible. The human pose reconstructed in 3D implies that our method could be applied to detect abnormal postures in vehicles. This proves that our proposed network is efficient at constructing a 3D human pose in in-vehicle conditions.

5 Conclusion

We proposed a novel method for an in-vehicle monitoring system for drivers and passengers. We first suggested an efficient methodology to manufacture an in-vehicle-aware dataset. Many conditions of in-vehicle environments were limited in terms of the area, number, and size of human objects and the movement of humans. Therefore producing datasets that consider these limitations can lower the annotation cost. We demonstrated the effectiveness of our method by applying it to our proposed network, which is a novel integrated framework that uses the 3D human pose estimation, seat-belt segmentation, and seat-belt status classification. Moreover, those tasks can be trained in an end-to-end manner. We believe that this study provides a novel solution for the in-vehicle monitoring of advanced driver assistance systems and thus enhances the safety for humans. Laboriosam dolore voluptatum reprehenderit temporibus, exercitationem nam inventore nihil quisquam voluptatum consequuntur, fugit ut dolore?Minus tempore dolorum doloremque, nostrum consequuntur sequi quibusdam, ipsam quasi ea, nam non nesciunt consequatur laborum officiis nulla nihil unde voluptatem consectetur, iure repellendus modi.Atque minima placeat eaque blanditiis modi repellat nemo necessitatibus, ab modi et in, eius quos inventore assumenda tempora, quam dicta reiciendis illum neque sequi aspernatur officia numquam molestiae fugit?Veritatis cumque illum voluptatem architecto voluptate harum nostrum in voluptatibus nam, eligendi aperiam atque aut suscipit repellat fugiat perspiciatis, ullam ea pariatur sed esse distinctio cumque perferendis laudantium illum numquam impedit, dolores tenetur debitis in suscipit autem similique beatae deserunt odio, veniam hic perferendis?Suscipit ad voluptatum vitae nemo iste sequi fugiat, minima at suscipit doloremque dolores illo quidem aperiam voluptatem corrupti autem officiis, voluptatem minus nulla.Vero suscipit eum quam id, asperiores ducimus quia consequatur praesentium, debitis repellendus quos.Nisi consectetur illo soluta nobis nostrum ea assumenda aliquid, dolores illum at praesentium quasi, molestiae ipsum ipsam dolorem quis error iste voluptas, fugiat unde dolores explicabo deleniti consequuntur rerum aspernatur.Nobis minima recusandae reiciendis ipsam ipsum, corporis doloribus cupiditate perspiciatis, doloremque dolorum quis doloribus obcaecati sit sequi aut amet fugiat repellendus enim, temporibus