A Real-time Face Detector on CPU Using Efficient Transformer

Muhamad Dwisnanto Putro, Adri Priadana, Duy-Linh Nguyen, and Kang-Hyun Jo
Department of Electrical, Electronic, and Computer Engineering, University of Ulsan
Ulsan, South Korea
Email: dputro@mail.ulsan.ac.kr; priadana3202@mail.ulsan.ac.kr; ndlinh301@mail.ulsan.ac.kr; acejo@ulsan.ac.kr

***Abstract*—Face detection is a basic vision method to find the facial location. It is usually used in the initial step of advanced facial analysis. Therefore, this approach is required to work quickly, especially on low-cost devices to support practical applications. A deep learning architecture can robustly extract the distinctive feature by employing a lot of weighted filters. However, the model produces heavy parameters and computational complexity. A transformer is a deep learning architecture that can capture the feature position relationship, which increases the detector performance. This work in this paper proposes a new efficient transformer architecture that is implemented to face detection. It can highlight the spatial information from a similarity map by utilizing a 2D-convolutional filter. This architecture generates low computation and lightweight trainable parameters that serve the proposed face detector to run fast on an inexpensive device. As a result, this proposed network achieves high performance and competitive precision with the low-cost model. Additionally, the proposed transformer module does not significantly add computation and parameters that can run fast at 95 frames per second on a Core i5 CPU.**

***Index Terms*—Efficient model, deep learning, face detection, transformer, real-time.**

I. Introduction

Face detection is a fundamental method in computer vision to find the location face area in an image. It is usually used as the initial process of the advanced method, such as facial expression, landmark, recognition identification, age, race, and gender classification [1]. Nowadays, portable technology needs this method installed inside the system [2]. Therefore, face detection demands work fast in real-time without ignoring the performance.

Feature extraction is the primary process of face detection to recognize specific facial features. This method captures distinctive elements that contain essential information about the human face, including the nose, eyes, lips, chin, eyebrows, and cheeks as unique components [3]–[5]. Those features have different shapes and textures, even though they are the same color. The relationship between features is also complex knowledge, containing rich information to distinguish a face from other objects.

Conventional feature extraction methods have been introduced to localize face elements using Haar-like features [6]. It finds the distinctive feature by subtracting dark and bright regions in the rectangle field. Furthermore, AdaBoost learns the candidate's facial features through a sequential classifier. Another work [7] has proposed a skin color method to find facial feature location using a probability model. It uses two extractors: Haar-like and Local Binary Pattern (LBP). It then uses the feature characters in the learning phase to boost features considered elements of the face. The traditional study showed satisfying efficiency in real-time speed. However, it constrains the precision that detects a small face, multi-pose, and complicated background.

The deep learning model has shown excellent results in extracting the distinctive feature [8]. It even can overcome a difficult challenge. This extractor can distinguish facial features by learning the specific element from the instance. A CNN-based architecture extracts spatial feature area by utilizing weighted filter [9]–[11]. However, it has a limitation on the relationship between global regions. Transformer architecture is used to tackle this issue by building global perspective correlation [12], [13]. It utilizes a self-attention module to capture the positional relationship of interest features. It then mixes the global information according to channel feature maps to enhance the selective feature stage. Nevertheless, this approach increases the number of parameters and uses heavy computation power. It obviously weakens the transformer architecture in the efficiency sector.

In this work, an efficient transformer architecture is offered to extract essential features at the end of the backbone. It uses spatial filter operation to capture the correlation features that compress the trainable parameter usage. This module also is assigned to enhance the informative feature through positional attention. Based on the discussion, the main contributions can summarize as follows:

1) A new efficient face detector is proposed to localize face regions that can overcome multi-pose, occlusion, and extreme background challenges.
2) A novel lightweight transformer module is presented to discriminate the valuable elements that comprehensively capture the positional correlation between features.
3) The efficiency of the detector is higher than other CPU detectors that achieve high precision on several benchmarks, including Annotated Faces in the Wild (AFW) [14], PASCAL face [15], and Face Detection Data Sets and Benchmarks (FDDB) [16].

This paper is arranged as follows: Section II presents the proposed architecture of face detection. Section III discusses the training and implementation setup. Section IV explains the

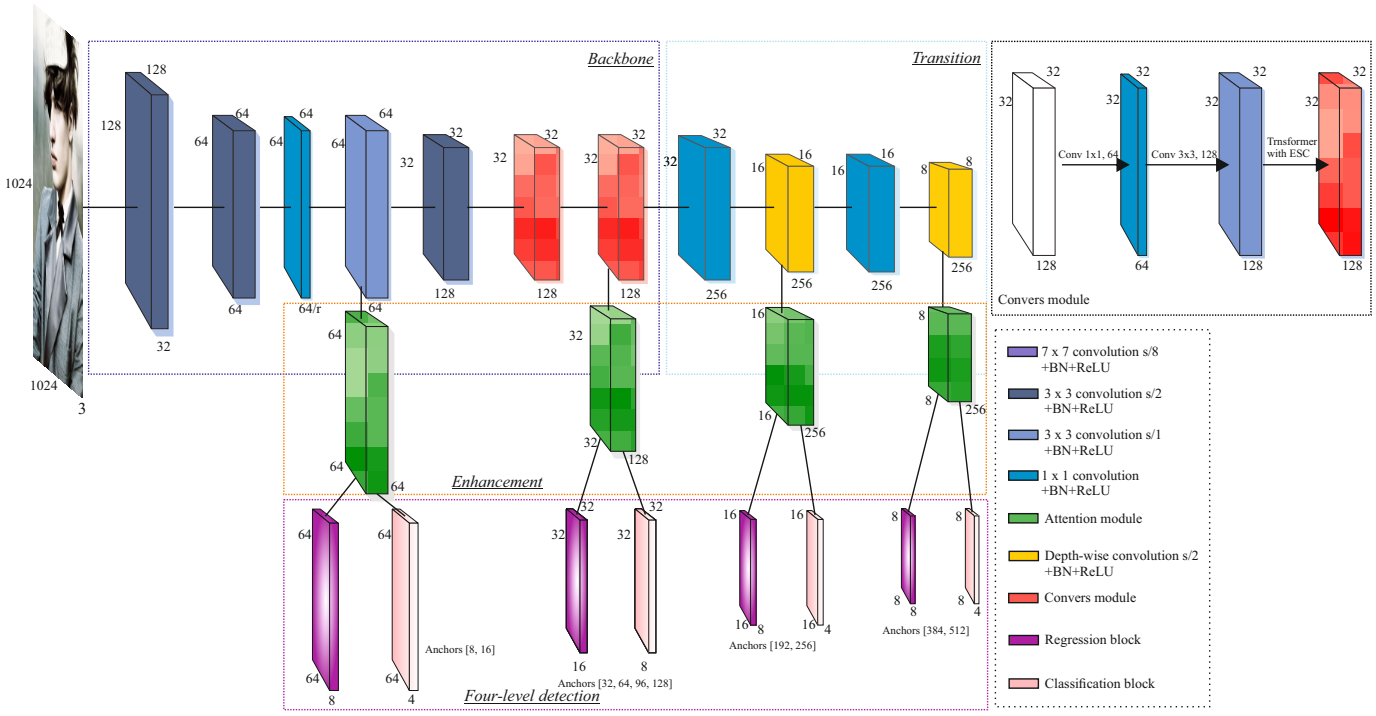


Fig. 1. The proposed network of face detection. This detector assigns backbone, transition, enhancement, and four detection layer to accurately find the face location. This model generates 532,000 parameters with 0.22 GFLOPS.

experiment and results. Finally, conclusions and future work are discussed in Section V.

II. Proposed Architecture

This section discusses the detector architecture in detail. The general architecture contains three vital modules: backbone, transition, enhancement, and prediction. It employs four-level detection to predict the face location on a different scale, as shown in Fig. 1.

A. Backbone with transformer

A backbone module plays an important role in discriminating the distinctive features that impact a detector's performance. The proposed face detector uses a shrinking approach to reduce the resolution size in the initial stage. Therefore, it utilizes sequential convolutional operation to catch out the spatial information. Therefore, It uses a 3×3 to shrink the input map, reducing the spatial scale. In addition, it uses a bottleneck module that utilizes a 1×1 and a 3×3 with a reducing channel in the initial operation. We claim it can save a number of parameters and computational complexity.

Furthermore, the proposed detector offers a convolutional with a transformer in a series block (Convers) to comprehensively capture the specific feature and enhance the relationship. It combines the bottleneck convolution module with an efficient transformer to increase the extractor performance. This proposed structure can produce lower parameters and computation than the standard transformer module [13], increasing the detector's efficiency. Fig. 2 shows that it uses positional encoding in the initial stage using a fully connected layer to remind the position of each element. This output can update the input features, which provides the weighted information of location features. In order to extract the features of the input attention block, it uses a convolutional operation using 5×1 kernel that is an efficient extractor to generate Query, Key, and Value components.

A dot product attention block is applied to find the similarity of position features between the Q and K map. It provides a high probability for elements with the same intensity from both maps. The similarity map can summarize the position information of the feature and update the value map that helps the model obtain valuable information accurately. Then, a spatial convolution with a ReLU activation filter out the end of the attention information. Instead of using Fully Connected in reconstruction stages, it offers efficient convolutional spatial-based that sequentially determines the narrow correlation area between each channel. It applies a 5×1 convolutional, following batch normalization and SiLU to prevent the vanishing gradient. Additionally, it uses a residual approach that transfers input features to the beginning and end of the reconstruction module. This technique can retain the output feature's quality by preventing the loss of information.

B. Transition module

The proposed detector uses a transition module to reduce the map resolution in medium and high-level features. It offers a cheap operation block by employing an inverted separable depthwise convolution block [17]. It contains a 1

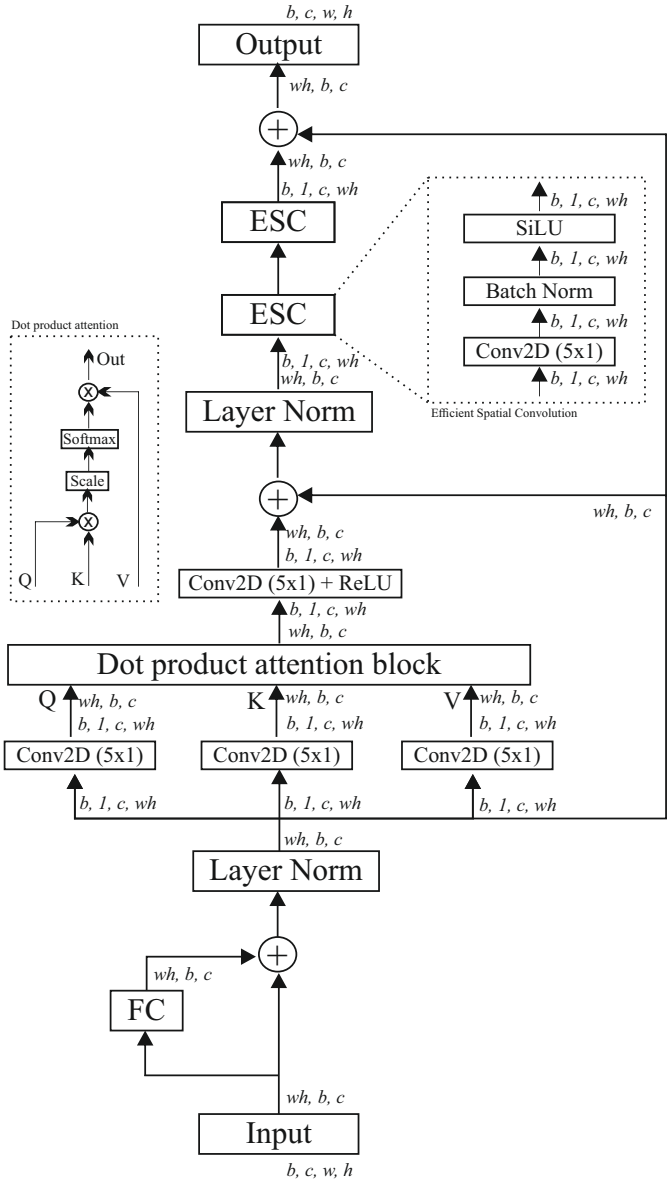


Fig. 2. The proposed efficient transformer using 2D-Convolutional filter

$\times$ 1 convolutional with a depthwise operation that generates a few parameters and operations.

C. Enhancement module

In order to enhance the specific features of each branch detection layer, we assign the squeeze excitation module. It adopts [18] work that operates global average pooling to obtain representative features according to the channel direction. The fully connected layer is used to construct the selective weight and then scale the input features.

D. Detection module

The proposed detector employs a multi-level prediction layer to estimate multiple faces location that accommodates various scale. It adopts the structure of the work [10] that utilize four-layer using different anchor scale. It assigns the

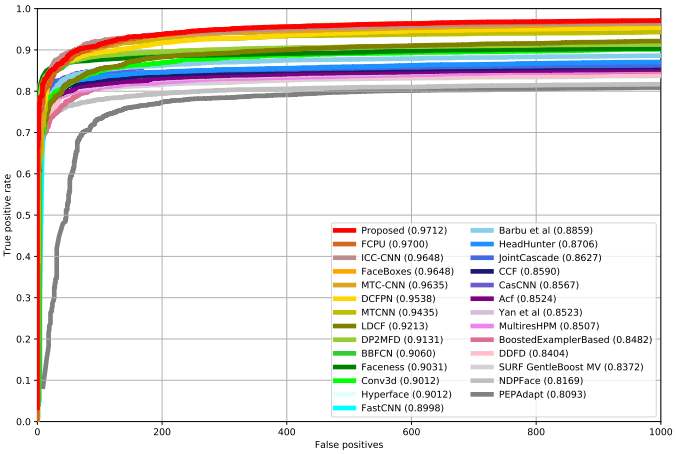


Fig. 3. Evaluation on the FDDB dataset using true postiove rate at 1,000 false positive.

first layer to predict tiny faces, the second layer to small faces, the third layer to medium faces, and the fourth layer to predict large faces. This structure effectively predicts various object scales by dividing the assignment of each layer according to the face sizes.

III. Training and Implementation Setup

The proposed face detector generates two output vectors, including regression scores (x, y, w, h) and probability class (face or none). This prediction is quantified by a Multi boxes loss [9] to measure the distance of the location and class probability prediction. This function contains regression and classification loss that utilize L1 smooth loss and Softmax loss, respectively. In the training phase, the WIDER FACE [19] is used for training knowledge with 12,800 pictures containing various challenges. It uses augmentation by random cropping, color manipulation, horizontal flipping, and resize transformation to create more instances for a robust training model. Additionally, we set several settings to optimize the training process. It applies Stochastic Gradient Descent (SGD) optimizer in updating the weighting process with the momentum of 0.9 and the weight decay of $5 \cdot 10^{-4}$. It defines batch size of 32 and multiple learning rates (10^{-3} - 10^{-5}) with 470 total epochs. In the evaluation stages, it sets IoU (Intersection over Union) is more than 0.5 to establish the prediction box. The proposed detector is simulated on a Pytorch application that uses a GTX1080Ti as an accelerator in the training phase. Besides, a Core i5-6600 with RAM of 8GB is used as the main CPU in the testing process.

IV. Experiments and Results

This section examines the detector's performance on several datasets and its efficiency on a CPU. It measures the efficiency cost and compares parameters, computation, and model speed to other CPU detectors.

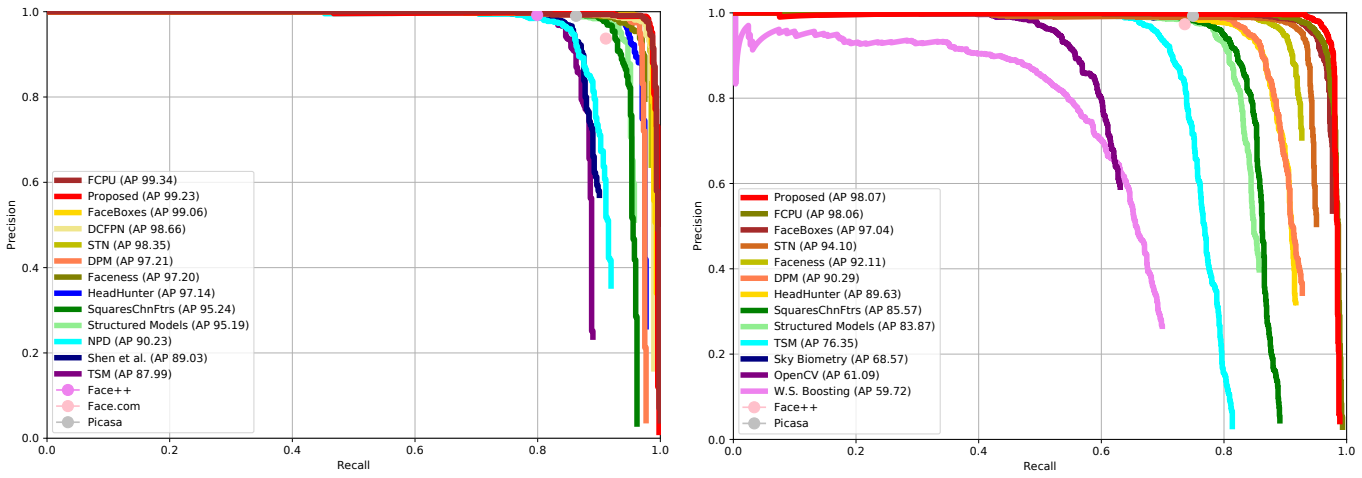
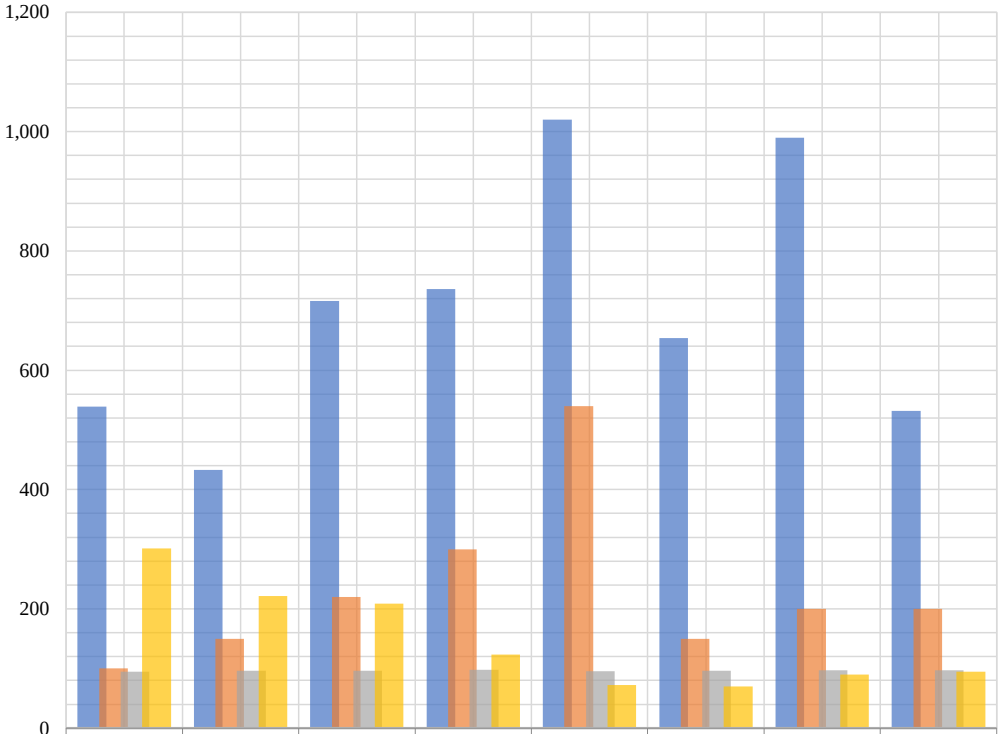


Fig. 4. Average precision results on AFW (a) and PASCAL face (b) dataset.



	LWFCPU	EFED	FFCPU	FAFCPU	DCFPN	FaceBoxes	FCPU	Proposed
Parameters (K)	539	433	716	736	1,020	654	990	532
MFLOPS	100	150	220	300	540	150	200	200
TPR on FDDB	94.40	96.50	96.60	97.80	95.40	96.65	97.00	97.12
FPS	301.26	221.37	208.98	123.12	72.57	70.23	90.24	94.85

Fig. 5. The efficiency comparison of CPU face detector. It examines the speed of model on Core i5-6600 CPU by live streams video.

A. *Evaluation on Benchmark*

1) FDDB dataset: The dataset provides 2,845 pictures with 5,171 annotated labels. Generally, the image collected is from the yahoo websites that cover a variety of challenges: multi-view, illuminance, and background complex. In this dataset, we use discrete categories by applying a rectangle bounding box to evaluate the prediction. The proposed detector obtains excellent performance and is superior to the leading competitor FCPU [9]. As shown in Fig. 3, it achieves 97% TPR (True Positive Rate), which also outperforms FaceBocxes and ICC-

CNN.

2) AFW dataset: The dataset provides fewer images than other datasets, which consist of 203 pictures. It uses 473 labels to evaluate instances that include several challenges, the same as the FDDB dataset. We use Average Precision (AP) in this evaluation dataset for a fair comparison with other detectors. Fig. 4 (a) illustrates that our detector achieves 99.23% AP. This performance is 0.11% lower than FCPU.

3) PASCAL face dataset: The dataset contains 851 picture that provides 1,335 annotated labels. It includes an indoor and outdoor environment that has complex features and backgrounds. In addition, this dataset is a subset of PASCAL VOC, containing single and multiple persons. Fig. 4 (a) shows that the proposed detector achieves 98.07% that outperforms other CPU detectors. It has slightly superior performance over FaceCPU, which differs by 0.01% AP.

B. Efficiency results

A CNN-based object detector tends to generate an abundance of parameters. It is due to the heavy use of convolution filter operations. It also impacts increasing computing power, which weakens the efficiency of the sector of a model. The proposed model results in a lightweight parameter of 532K with 0.2 GFLOPS. This result shows that our proposed detector is more lightweight than FFCPU [4], FAFCPU [10], FCPU [9], FaceBoxes [8], and DCFPN [20]. Although EFED and LWFCPU use lower computational costs than our model, these competitors did not achieve high precision. Fig. 5 illustrates the results of the proposed detector's efficiency and its comparison with other low-cost face detectors.

Furthermore, the proposed detector is evaluated its speed when implemented on a CPU. It measures the ability of the detector to work on low-cost devices by computing the model running time. This experiment uses a Core i5-6600 CPU without a graphics accelerator supporting device, which increases the capability of practical applications. The proposed model can smoothly operate at 94.85 FPS, which is faster than other CPU detectors, including DCFPN, FaceBoxes, and FCPU. Even though LWFCPU, EFED, and FFCPU are faster than our model, they have low performance. On the other hand, FAFCPU has high precision and speed, but the number of parameters and computations of this detector is higher than the proposed detector.

V. Conclusion

This paper presents a CPU-based face detector that uses an efficient transformer module to improve its performance. The proposed transformer model can capture the relationship of global features that build broader representative information. It also can highlight the contextual element, improving important information that impacts correct prediction decisions. As a result, the proposed detector achieves excellent accuracy that outperforms the FCPU detector on FDDB and PASCAL face datasets. Regarding efficiency, the proposed model produces less computation and parameters than this competitor. It can smoothly run at 95 FPS on a Core i5 CPU. In future work, the loss function exploration will be carried out to improve performance without reducing the inference speed.

Acknowledgment

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the government (MSIT). (No.2020R1A2C200897212).

References

[1] Z. Zhao, Q. Liu, and S. Wang, "Learning deep global multi-scale and local attention features for facial expression recognition in the wild," *IEEE Transactions on Image Processing*, vol. 30, pp. 6544–6556, 2021.

[2] M. D. Putro and K. Jo, "Real-time face tracking for human-robot interaction," in *2018 International Conference on Information and Communication Technology Robotics (ICT-ROBOT)*, Sep. 2018, pp. 1–4.

[3] M. D. Putro, D.-L. Nguyen, and K.-H. Jo, "Lightweight convolutional neural network for real-time face detector on cpu supporting interaction of service robot," in *2020 13th International Conference on Human System Interaction (HSI)*, 2020, pp. 94–99.

[4] M. D. Putro and K.-H. Jo, "Fast face-cpu: A real-time fast face detector on cpu using deep learning," in *2020 IEEE 29th International Symposium on Industrial Electronics (ISIE)*, 2020, pp. 55–60.

[5] C. Wang, J. Xue, K. Lu, and Y. Yan, "Light attention embedding for facial expression recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 4, pp. 1834–1847, 2022.

[6] P. Viola and M. J. Jones, "Robust real-time face detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, May 2004. [Online]. Available: https://doi.org/10.1023/B:VISI.0000013087.49260.fb

[7] Y. Ban, S.-K. Kim, S. Kim, K.-A. Toh, and S. Lee, "Face detection based on skin color likelihood," *Pattern Recognition*, vol. 47, no. 4, pp. 1573 – 1585, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S003132031300455X

[8] S. Zhang, X. Wang, Z. Lei, and S. Z. Li, "Faceboxes: A cpu real-time and accurate unconstrained face detector," *Neurocomputing*, vol. 364, pp. 297 – 309, 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0925231219310719

[9] M. D. Putro, L. Kurnianggoro, and K.-H. Jo, "High performance and efficient real-time face detector on central processing unit based on convolutional neural network," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 7, pp. 4449–4457, 2021.

[10] M. D. Putro, D.-L. Nguyen, and K.-H. Jo, "An efficient face detector on a cpu using dual-camera sensors for intelligent surveillance systems," *IEEE Sensors Journal*, vol. 22, no. 1, pp. 565–574, 2022.

[11] M. D. Putro, Duy-Linh, and K.-H. Jo, "Efficient face detector using spatial attention module in real-time application on an edge device," in *Intelligent Computing Theories and Application*. Cham: Springer International Publishing, 2021, pp. 829–841.

[12] F. Ma, B. Sun, and S. Li, "Facial expression recognition with visual transformers and attentional selective fusion," *IEEE Transactions on Affective Computing*, pp. 1–1, 2021.

[13] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations*, 2021. [Online]. Available: https://openreview.net/forum?id=YicbFdNTTy

[14] X. Zhu and D. Ramanan, "Face detection, pose estimation, and landmark localization in the wild," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2012, pp. 2879–2886. [Online]. Available: https://www.ics.uci.edu/~xzhu/face/

[15] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, Jun. 2010. [Online]. Available: http://host.robots.ox.ac.uk/pascal/VOC/voc2012/

[16] V. Jain and E. Learned-Miller, "Fddb: A benchmark for face detection in unconstrained settings," University of Massachusetts, Amherst, Tech. Rep. UM-CS-2010-009, 2010. [Online]. Available: http://vis-www.cs.umass.edu/fddb/index.html

[17] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4510–4520.

[18] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-excitation networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 8, pp. 2011–2023, 2020.

[19] S. Yang, P. Luo, C. C. Loy, and X. Tang, "Wider face: A face detection benchmark," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 5525–5533. [Online]. Available: http://mmlab.ie.cuhk.edu.hk/projects/WIDERFace/index.html

[20] S. Zhang, X. Zhu, Z. Lei, X. Wang, H. Shi, and S. Z. Li, "Detecting face with densely connected face proposal network," *Neurocomputing*, vol. 284, pp. 119–127, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925231218300274