

Face Expression Recognition via Transformer-Based Classification Models

M. Cihad Arslanoglu, Huseyin Acar and Abdulkadir Albayrak


Abstract—Facial Expression Recognition (FER) tasks have widely studied in the literature since it has many applications. Fast development of technology in deep learning computer vision algorithms, especially, transformer-based classification models, makes it hard to select most appropriate models. Using complex model may increase accuracy performance but decreasing inference time which is a crucial in near real-time applications. On the other hand, small models may not give desired results. In this study, it is aimed to examine accuracy and data process time performance of 5 different relatively small transformer-based image classification algorithms for FER tasks. Used models are vanilla Vision Transformer (ViT), Pooling-based Vision Transformer (PiT), Shifted Windows Transformer (Swin), Data-efficient image Transformers (DeiT), and Cross-attention Vision Transformer (CrossViT) with considering their trainable parameter size and architectures. Each model has 20-30M trainable parameters which means relatively small. Moreover, each model has different architectures. As an illustration, CrossViT focuses on image using multi-scale patches and PiT model introduces convolution layers and pooling techniques to vanilla ViT model. Model performances are evaluated on CK+48 and KDEF datasets that are well-known and most used in the literature. It was observed that all models exhibit similar performance with literature results. PiT model that includes both Convolutional Neural Network (CNN) and Transformer layers achieved the best accuracy scores 0.9513 and 0.9090 for CK+48 and KDEF datasets, respectively. It shows CNN layers boost performance of Transformer based models and help to learn data more efficiently for CK+48 and KDEF datasets. Swin Transformer performs 0.9080 worst accuracy score for CK+48 dataset and 0.8434 nearly worst score for KDEF dataset. Swin Transformer and PiT exhibit worst and best image processing performance in terms of spent time, respectively. This makes PiT model suitable for real-time applications too. Moreover, PiT model require 25 and 83 second least training epoch to reach these performance for CK+48 and KDEF, respectively.


Index Terms—FER, Transformers, ViT, Classification


I. INTRODUCTION

FACIAL expressions are universal communication skills independent of country, language or ethnicity for the people. Human facial mimics convey a rich information about emotions, behaviors, and so on [1]. Many disciplines study

facial expressions such as psychology [2] and marketing [3]. Tang et al. proposed a real-time system to evaluate the performance of students in a classroom using their facial expressions [4]. Sajjad et al. designed a framework to detect suspicious persons using facial expressions for smart security in law enforcement services [5]. Fu et al. evaluated whether there is a relation between being depressive and facial expressions. They showed that depressive people have poor ability to imitate facial expressions [6]. Since facial expressions play a significant role in many fields, Facial Expression Recognition (FER) has been of keen interest in computer vision and machine learning. Humans represent their facial expressions in many forms like being happy, less happy, or happier. In computer vision and machine learning, however, they are restricted in 6-8 essential form. Ekman and Friesen proposed to use 6 basic forms of emotions, which are anger, disgust, fear, happiness, sadness and surprise [7]. There are many techniques for FER tasks in the literature. However, all of them are categorized in two groups: geometric and appearance-based feature extraction algorithms. While geometric-based methods utilize the face landmarks of humans, appearance-based algorithms use texture, shape, and color-based features. Sheth et al. stated that geometric-based features outperform appearance-based feature extraction techniques [8]. Moreover, combining both appearance and geometric based features improve accuracy for FER tasks [9], [10], [11]. In geometric feature extraction approaches, the aim is to find a relationship between human mimics and facial expressions [12]. Geometric-based FER applications have two steps: face landmark detection and facial expression recognition. In face landmark detection stage, mimic points like eyes, mouth, and nose are detected and many statistical features are calculated such as distance between eyebrow and area of mouth. After the detection stage, obtained features are fed into a classifier to detect facial expression. In appearance-based FER algorithms, texture, color, and shape information are extracted from human face. There are two kinds of appearance-based feature extraction algorithms: deep learning and hand-crafted features. These two approaches have advantages and disadvantages. Hand-crafted feature extraction algorithms are easy to implement in terms of time and complexity. However, they have lack of adaptiveness for different conditions such as environment, human ethnicity, and so on. On the other hand, deep learning algorithms are able to learn these kinds of different cases. This strong learning ability comes with greater complexity. Deep learning methods need more computational power, data and time to yield competitive results. One of the most used deep learning algorithms in computer vision and FER tasks is CNNs. CNNs apply n_{xn}

 **M.Cihad ARSLANOĞLU** is with the Department of Electrical and Electronics Engineering, Engineering Faculty, Dicle University, Diyarbakir, 21280 TURKEY e-mail: cihatdt.21@gmail.com

 **Hüseyin ACAR** is with the Department of Electrical and Electronics Engineering, Engineering Faculty, Dicle University, Diyarbakir, 21280 TURKEY e-mail: hacar@dicle.edu.tr

 **Abdülkadir ALBAYRAK** is with the Department of Computer Engineering, Engineering Faculty, Dicle University, Diyarbakir, 21280 TURKEY e-mail: kadir.albayrak@dicle.edu.tr

Manuscript received May 18, 2024; accepted August 20, 2024. DOI:10.17694/bajece.1486140

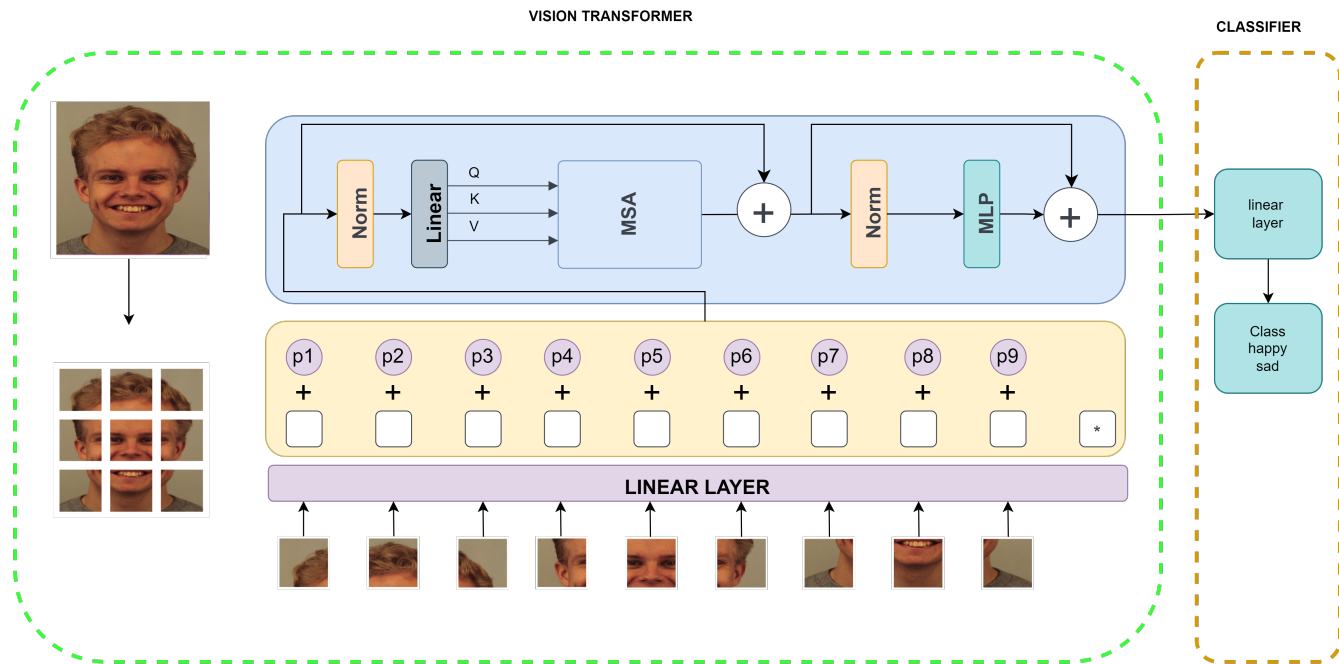


Fig. 1: High level architecture of ViT algorithm. ViT algorithm splits an image into equal sized patches. To calculate global embeddings and reduce data size, a linear layer is applied to each patch. Positional encoding values are added to each patch due to emphasize importance of their positions. These embeddings are given to multi-head attention mechanism and MLP block which includes linear layer and GeLU activation function. At the last stage, embeddings are given to a classifier which is a linear layer.

filters to input images to capture meaningful local features like shape, color, and texture. It uses pooling layers to decrease dimensions of input image to eliminate useless features and decrease computational complexity. CNNs provide more representative and less complex feature vector at the last layer. A classifier is fed with this obtained feature vector to detect facial expression. Vision Transformer (ViT) [13] is a Transformer-based, which is proposed for Natural Language Processing (NLP) task by Vaswani et al. [14], classification algorithm. ViT algorithm focuses on global dependencies of the images unlike CNNs. ViT algorithm divides an image into patches instead of processing it as a whole. Linear layers are applied to these patches to capture most representative features of image and decrease size of raw data. Then, a similarity metric, which is cosine similarity, is calculated between these all patches which is named as attention map. ViT algorithm uses this attention map to eliminate less valuable features like noise and gives a feature vector at the end of the processes. Obtained feature vectors are given to a classifier to detect facial expression of human. Although ViT algorithm outperform CNN based methods, it has a couple of disadvantages. ViT algorithm has $O(N^2)$ computation complexity which makes ViT training harder and time inefficient. Anasosalu et al. introduce a new token mixing operator, which is called RepMixer, to solve time latency problem without accuracy reduction [15]. Liu et al propose a hierarchical shifting window to capture non-overlapping features and local features with linear complexity increment. In this paper, FER accuracy and spent time performance of five different ViT models which are vanilla ViT [13],

Swin Transformer [16], CrossViT[17], PiT [18] and Deit [19] were compared. The performances of these used models are evaluated on two different datasets that are CK+48 and KDEF. The main contributions of this study are listed below:

- Accuracy and image process time performance of five different transformer-based models were compared.
- Each experiments were repeated five times to be sure about performance consistency and experiment reproducibility.
- Loss function plots and accuracy score of used models are evaluated due to reveal overfitting problem.
- Four different evaluation metrics were used to be sure about the results and see if there is a problem about results such as unbalanced data issue.
- Obtained results were compared with the literature.

II. RELATED WORKS

Related works were split into two subtitle that are hand-crafted methods and deep learning approaches. Hand-crafted methods also have different approaches such as appearance and geometric-based feature extraction. However, they were examined under same title that is hand-crafted methods.

A. Hand-crafted Methods

In [20], [21] geometric features are extracted and given to Hidden Markov Model (HMM) and Support Vector Machine (SVM) to classify. Rahul et al. achieves 84.7% accuracy score for Japanese Female Facial Expression (JAFFE) dataset [20].

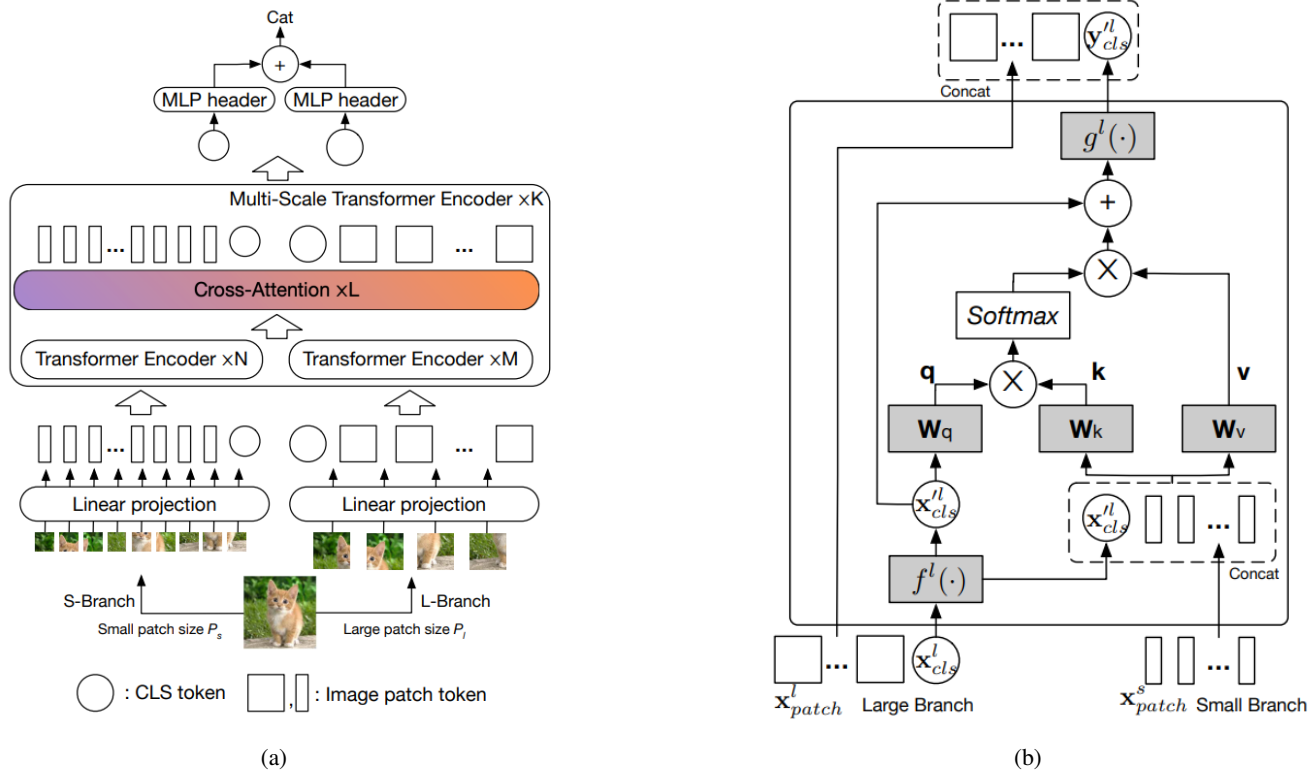


Fig. 2: (a) high level architecture of CrossViT algorithm. (b) high level architecture of cross-attention fusion approach [17]

Chouhayebi et al. states that they obtained 94.5% accuracy score for fusion of private and Bogazici University Head Motion Analysis Project (BUHMAP) datasets [21]. Sharma et al. apply preprocessing Gaussian filter for noise removal and contrast histogram equalization for illumination correction before detecting geometric features and classification [22]. Sharma et al. accomplish 95.5% accuracy score for Multi-media Understanding Group (MUG) dataset. Ibrahim et al. crop face of image that helps to extract most representative features. Then Histogram of Oriented Gradient (HOG) and Local Binary Pattern (LBP) algorithms are applied to extract features of the face. At the classification stage, obtained feature vectors are fed into SVM classifier [23]. Ibrahim et al. achieves 95.17% accuracy score for JAFFE dataset. Kaya et al. align face of image Generalized Procrustes Analysis (GPA) to obtain good face representation. Then many feature extraction methods like HOG, LBP Speedup Robust Features (SIFT), Local Phase Quantization (LPQ) are applied to aligned images. At the classification stage, two classification algorithms which are Extreme Learning Machine (ELM) and Partial Least Squares (PLS) are trained and their decisions are fused in the test stage [24]. The proposed method performs 53.62% accuracy score EmotiW 2015 dataset.

B. Deep Learning Approaches

Liu et al. gives a video to CNN and Global Attention Unit to extract spatial features and use Bidirectional Long Short-Term Memory (BiLSTM) to capture temporal variations from previous layers. At last stage, Attention pooling is applied and

given to classification layer [25]. In the paper, it is stated that the proposed algorithm exhibits 99.54%, 88.33%, 87.06%, and 63.71% accuracy score performance for CK+48, OuluCASIA, MMI, and AffectNet, respectively. Pan et al. extract spatial and temporal features giving frames of a video to two model stacks which consist of CNN and Long Short-Term Memory (LSTM) models sequentially. Obtained feature vectors from two models stacks are aggregated using proposed aggregation layer. A SoftMax activation layer is applied to aggregated feature vectors and classified [26]. Pant et al. declare that proposed method achieves 65.72% and 42.98% accuracy scores for RML and eNTERFACE datasets. Uddin et al. uses depth cameras to capture images instead of conventional RGB cameras. They train a CNN based classification algorithm using hand-crafted features like Local Directional Rank Histogram Pattern (LDRHP), Local Directional Strength Pattern (LDSP), and Generalized Discriminant Analysis (GDA) [27]. Uddin et al. obtained 95.42% and 96.25% accuracy scores for CK and Bosphorus datasets. Minaee et al. states that there is no need very deep CNN models for FER tasks and propose two CNN model that has four and two convolution layers, respectively. Second CNN model is named as localization network. An affine transform applied to feature vectors that is obtained by localization network and multiplied with feature vectors of first model. In the classification stage, a linear layer is applied to the feature vector and classified using linear layer and SoftMax activation function [28]. Minaee et al. achieves 70.02%, 98.0%, 99.3%, and 92.8% accuracy scores for Facial Expression Recognition 2013 (FER2013), CK+48, Facial

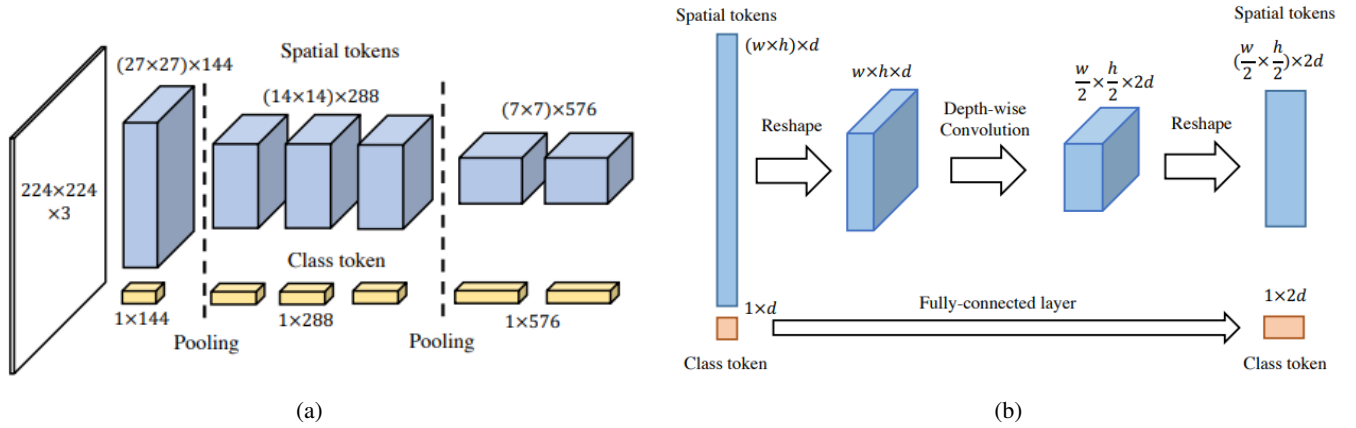


Fig. 3: (a) shows high-level architecture of PiT algorithm. In (b), PiT pooling layer is displayed [18]

Expression Research Group (FERG), and JAFFE datasets.

III. MATERIAL AND METHODS

ViT is a transformer-based image classification algorithm. Dosovitskiy et al. proposed ViT algorithm by inspiring attention mechanism based transformer architecture which is designed for NLP by Vaswani et al. [14]. ViT algorithm has three main stages: patch embedding, attention mechanism and classification. In patch embedding, an image with $W \times H \times C$ size is shaped into $L \times (n \times n) \times C$ where L is patch number, n is patch size, and C is channel size. A linear layer is applied to each to obtain embeddings and reduce patch size. To specify where each patch belongs to, positional encoding values are added to output of linear layer. In attention mechanism, attention matrix, cosine similarity between all patch embeddings, is calculated. Then attention matrix is multiplied by embeddings to weight embeddings. A MLP block which has linear layer and Gaussian Error Linear Unit (GeLU) activation function is applied to embeddings due to enhance embeddings. As a result, these obtained embeddings are given to a classifier at the classification stage. Figure 1 shows high level architecture of ViT algorithm.

A. CrossViT

Vanilla ViT model requires many data for efficient training since it focuses on global dependencies. CrossViT algorithm focuses on both global and local embedding tokens by using two level patch size. In patch embedding phase, CrossViT algorithm splits the image into $m \times m$ and $n \times n$ sized patches. These patches are given to cross-attention module that includes a fusion approach to combine both different sized patches and reduce computation complexity. CrossViT algorithm and cross-attention illustrations are given in Figure 2a and 2b, respectively.

B. PiT

Heo et al. state that adding spatial dimension reduction in ResNet CNN-based deep learning algorithm improves accuracy score and decrease validation loss in training stage [18].

ViT does not have any spatial reduction layer unlike CNN-based algorithms. PiT algorithm uses a pooling layer to utilize advantages of spatial dimension reduction. Proposed pooling layer apply a couple of depth-wise convolution operation in ViT's patch embedding stage and obtain 3D tensors unlike 2D matrix like vanilla ViT. These 3D tensors are reshaped into 2D matrix before transformer architecture and ViT procedure is sustained. High-level architecture of PiT and pooling layer were provided in Figure 3a and 3b, respectively.

C. DeiT

ViT algorithm needs million-level images to learn image representation embeddings efficiently. This makes ViT hard learner and hardware inefficient deep learning model. Touvron et al. proposes a distillation token and label distillation techniques with using teacher-student relation in order to decrease these data and hardware requirements [19]. In DeiT architecture, a distillation token is concatenated with patch embeddings. In backpropagation, layer teacher and student model decisions are combined using soft distillation and hard-label distillation approaches like in Equation 1 and 2, respectively. In Equation 1 and 2, \mathcal{L}_{CE} is cross-entropy, KL is Kullback-Leibler (KL) loss, λ is coefficient to balance KL and cross-entropy, ψ is softmax function, Z_s is student model logits, τ is distillation temperature, Z_t is teacher model logits, and y_t is decision of teacher model.

$$\mathcal{L}_{global} = (1 - \lambda) \cdot \mathcal{L}_{CE}(\psi(Z_s), y) + \lambda \cdot \tau^2 \cdot KL\left(\psi\left(\frac{Z_s}{\tau}\right), \psi\left(\frac{Z_t}{\tau}\right)\right) \quad (1)$$

$$\mathcal{L}_{global}^{hardDistill} = \frac{1}{2} \cdot \mathcal{L}_{CE}(\psi(Z_s), y) + \frac{1}{2} \cdot \mathcal{L}_{CE}(\psi(Z_s), y_t) \quad (2)$$

D. Swin Transformer

Since ViT focuses on all global dependencies between each patch, it has N^2 computational complexity where N is patch size. Moreover, ViT does not focus on hierarchical relations in the image. Liu et al. proposes Swin transformer with shifted

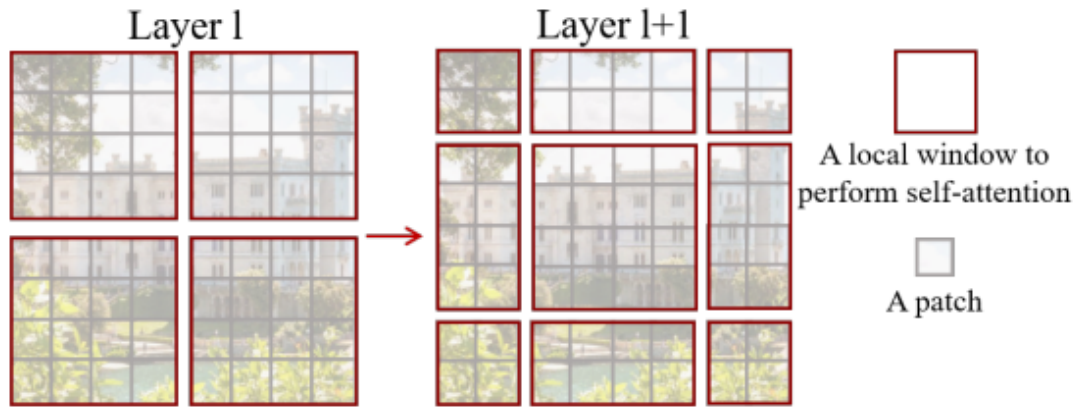


Fig. 4: Non-overlapping shifted window architecture [16]

non-overlapping window and Swin transformer block to reduce computation complexity and reveal hierarchical relations in the image [16]. Given input image is split into patches like ViT but with small patch size like 4×4 . Linear layer is applied to each patch with C output size and obtained $\frac{H}{4} \times \frac{W}{4}$ times patch embeddings. In order to reduce attention mechanism computation complexity, a modified attention mechanism, called Swin Transformer Block, is used to calculate relations between patches. Swin transformer block apply conventional transformer block for only some patches that is in a non-overlapping window. Non-overlapping window includes M patches where M is initialized as 4. In the other words, each non-overlapping window possess $4C$ times patch embeddings. This proposed Swin transformer block reduce computation complexity from $4hwC^2 + 2(hw)^2 C$ to $4hwC^2 + 2M^2hwC$ where h and w are height and width of the image, M is patch size in the non-overlapping window, and C is patch embedding size. Later on, merging layer is applied to each window separately. Merge layer concatenate each patch in the non-overlapping window and apply linear layer in order to decrease embedding size to $2C$ from $4C$. This sequential process is applied to n times to output of each swin transformer block with shifting non-overlapping window as illustrated in Figure 4. High-level architecture of Swin transformer is shown in Figure 5.

IV. RESULTS AND DISCUSSION

A. Dataset

To evaluate transformer based five different classification algorithms, two well-known and commonly used datasets that are Karolinska Directed Emotional Faces (KDEF) and Extended Cohn-Kanade (CK+48) were used. Both datasets were split into train and test datasets with 80% and 20% rate, respectively.

1) *KDEF*: KDEF dataset is provided to literature by Karolinska Institute in 2008 [29]. KDEF dataset includes 4900 images that is taken from 35 male and 35 female individuals in laboratory conditions. Each class have same sample size that is 700. Obtained images has same shape and it is $562 \times 762 \times 3$ where 562 is width, 762 is height and 3 is channel size. Dataset

has 7 emotion state: afraid, angry, disgusted, sad, happy surprised, and neutral. Captured image is taken in 5 different views: full left profile, half left profile, straight, half right profile, and full right profile. Sample images with different views from KDEF dataset were illustrated in Figure 6. AF, AN, DIS, HAP, NEU, SAD, and SUP abbreviations represent afraid, angry, disgusted, happy, neutral, sad, and surprised, respectively. FLP, HFP, FSP, HRP, and FRP indicates full left profile, half left profile, full straight profile, half right profile, and full right profile.

2) *CK+48*: CK+48 dataset is created by Luckey et al. in 2010 as extension of Cohn-Kanade dataset [30]. The dataset contains 981 images with $48 \times 48 \times 3$ shape and seven classes. Class names their sample sizes are following 75 afraid, 135 angry, 177 disgusted, 84 sad, 207 happy, 249 surprised, and 54 neutral. Sample images from CK+48 were given in Figure 7. AF, AN, DIS, HAP, NEU, SAD, and SUP abbreviations represent afraid, angry, disgusted, happy, neutral, sad, and surprised, respectively.

B. Setup

All experiments were done using Python programming language and PyTorch [31] deep learning framework. A couple of important parameters such as input size and patch size were shared in Table I. Input column represents input size of images. Patch column indicates how many patches are extracted from the input image. CrossViT has two patch size since it uses multi-scale patches. Total model trainable parameter numbers and their architectures were cared to select transformer-based classification algorithms. Table I 'params' column display total trainable parameters size of models. 'img/sec [GPU]' and 'img/sec [CPU]' columns mean that how many images are processed in one second with GPU and CPU hardware, respectively. Timm implementation of used deep learning models that is pretrained on Imagenet1k dataset were fine tuned instead of from scratch training. ImageNet1k dataset has 1000 classes with over 1 million images. All models were trained until train accuracy reaches up to 98%. The average number of epochs needed to achieve 0.98 train accuracy score is shared in 'epoch' column of Table II. CrossViT, DeiT, PiT, Swin

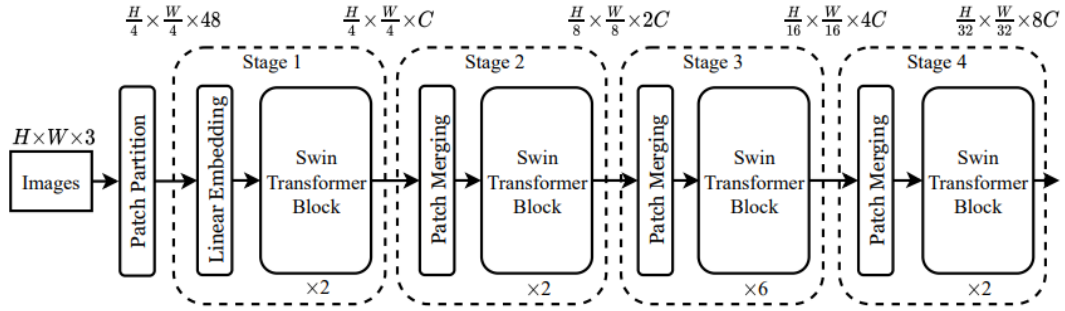


Fig. 5: High-level architecture of Swin Transformer[16]

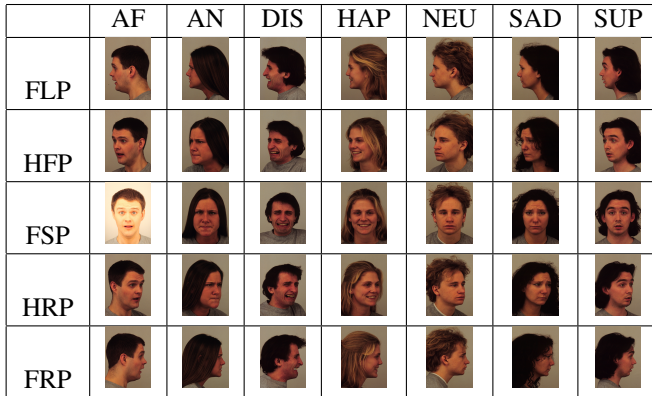


Fig. 6: Sample images from KDEF dataset.

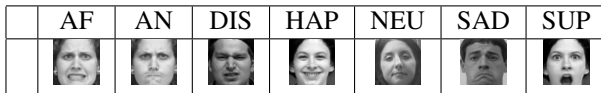


Fig. 7: Sample images from CK+48 dataset.

TABLE I: Parameters of used models

Model	Input	Patch	Embeddings	Head	Depth	params (M)	img/sec [GPU]	img/sec [CPU]
CrossViT	240x240	(12,16)	(192,384)	(6,6)	(1,4,0)	26.9	red	5
DeiT	224x224	16	384	6	12	22.0	120	6
PiT	224x224	16	(48,48,48)	(3,6,12)	(2,6,4)	23.5	160	9
Swin	256x256	16	96	(3,6,12,24)	(2,2,6,2)	28.3	48	2
ViT	224x224	16	192	3	12	22.1	119	6

Transformer, and ViT algorithms achieved a training accuracy of 98% after approximately 12, 32, 25, 93, and 42 epochs, respectively, on the CK+48 dataset. When applied to the KDEF dataset, the ordered algorithms reached the same 98% accuracy after 15, 105, 83, 100, and 106 epochs, respectively. Stochastic Gradient Descent (SGD) optimization algorithm was used with linearly decreased learning rate from 0.1 to 0.01 throughout first 30 epochs. It was observed that training the models with constant learning rate increase training time to reach 98% train accuracy. Batch size was set to 32 for all experiments. All models are trained and tested on Google Colab environment with following hardware specifications: 16GB Random Access Memory (RAM), Intel(R) Xeon(R) CPU @ 2.20GHz and Tesla T4 GPU with 16gb Memory. All experiments were done five times to be sure about consistency and reproducibility of the results.

1) *Evaluation Metrics*: Four metrics that are accuracy, precision, recall, and F1 were used in order to compare performance of used models. Definition of used evaluation metrics were given in Equation 3, 4, 5, and 6. Accuracy score measure how many data is correctly predicted by the model. It is calculated dividing True Positive (TP) plus True Negative (TN) to sum of True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) like in Equation 3. A high accuracy score indicates the model is capable to classify data correctly. Although accuracy score is a suitable metric to evaluate balanced datasets, it is specious when data is not balanced.

$$A = \frac{TP + TN}{TP + TN + FP + FN} \quad (3)$$

Unlike accuracy score, precision focuses on how much model is well to predict target, TP, instead of others like TN. Precision is not affected by unbalanced datasets since it does not focus on other classes except target one. It is calculated as in Equation 4.

$$P = \frac{TP}{TP + FP} \quad (4)$$

Recall score does not take account FP predictions which means how many data is predicted as target class when they are not target class. Its calculation formula was given in Equation 5.

$$R = \frac{TP}{TP + FN} \quad (5)$$

F1 combines both recall and precision score and yields a single value. Since F1 is a tradeoff between recall and precision scores, it provides a more reliable result. It is calculated like in Equation 6.

$$F1 = 2 \cdot \frac{P \cdot R}{P + R} \quad (6)$$

Table I 'img/sec [GPU]' and 'img/sec [CPU]' columns reveal that PiT and Swin Transformer models has most and least image processing capability in a second for GPU and CPU, respectively. While PiT process 160 images in a second, Swin Transformer has 48 image processing capability on GPU hardware. On the other hand, same models process 9 and 2 images in a second for CPU hardware.

Obtained average performance metrics of five different experiments were shared in Table II for both CK+48 and

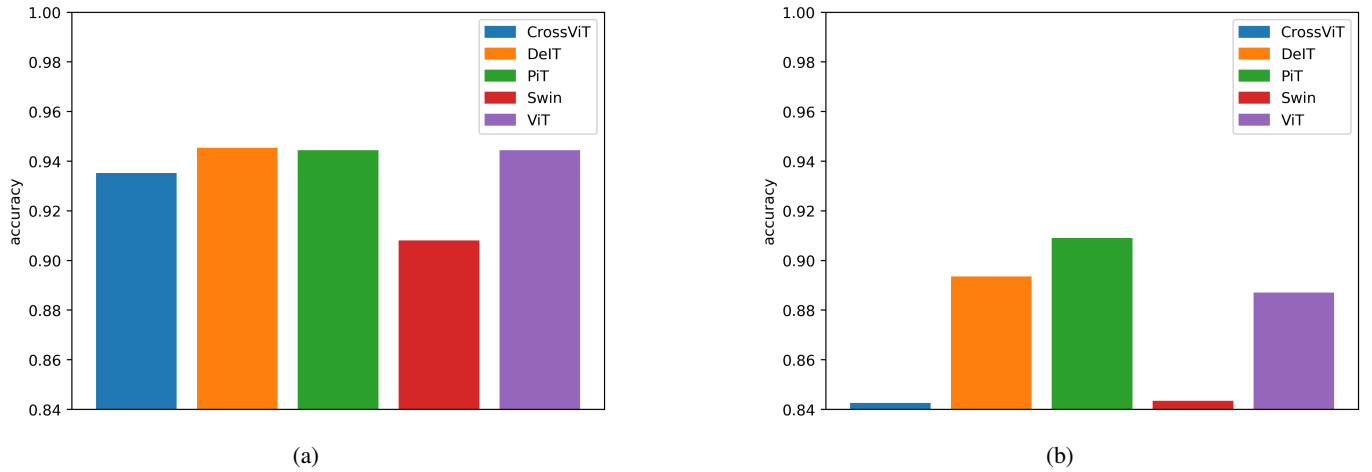


Fig. 8: (a) and (b) represents accuracy scores of CK+48 and KDEF test datasets, respectively.

TABLE II: Performance of models for both CK+48 and KDEF datasets.

	CK+48					KDEF				
	Accuracy	Precision	Recall	F1	epoch	Accuracy	Precision	Recall	F1	epoch
CrossViT	0.9463±0.027	0.9459±0.029	0.9465±0.031	0.9454±0.031	12	0.8426±0.011	0.8447±0.010	0.8426±0.011	0.8426±0.011	15
DeiT	0.9400±0.020	0.9389±0.026	0.9389±0.019	0.9394±0.025	32	0.8935±0.005	0.8944±0.005	0.8936±0.005	0.8935±0.0056	105
PiT	0.9513±0.023	0.9504±0.030	0.9498±0.021	0.9514±0.029	25	0.9090±0.012	0.9101±0.012	0.9091±0.012	0.9089±0.0125	83
Swin	0.9080±0.042	0.8869±0.050	0.8843±0.045	0.9076±0.050	93	0.8434±0.004	0.8449±0.005	0.8435±0.046	0.8428±0.0050	100
ViT	0.9444±0.004	0.9429±0.009	0.9424±0.012	0.9440±0.007	42	0.8871±0.010	0.8871±0.011	0.8872±0.010	0.8866±0.011	106

TABLE III: Studies from the literature

Study	Dataset	Accuracy
Wang et. al [32]	CK+48	0.9284
Subud et. al [33]	CK+48	0.9987
Kim et. al [11]	CK+48	0.9646
Yu et. al [34]	CK+48	0.9410
Hu et. al [35]	CK+48	0.9407
Mohan et. al [36]	CK+48	0.9800
Kumar et. al [37]	CK+48	0.9420
Kas et. al [38]	CK+48	0.9648
This study	CK+48	0.9513
Subud et. al [33]	KDEF	0.9689
Eng et. al [39]	KDEF	0.8095
Puthanidam et. al [40]	KDEF	0.8958
Mohan et. al [36]	KDEF	0.8300
Kumar et. al [37]	KDEF	0.9370
Obait et. al [41]	KDEF	0.9529
Kas et. al [38]	KDEF	0.9020
Yaddaden et. al [42]	KDEF	0.8458
Barra et. al [43]	KDEF	0.8271
This study	KDEF	0.9090

KDEF datasets. The notation in Table II is that accuracy $\pm \sigma$ where σ is standard deviation of 5 experiment. It is clearly seen standard deviation of all results are less than 0.06 which means results are consistent. The best results for both CK+48 and KDEF datasets are obtained by PiT transformer architecture. PiT architecture achieves average best 0.9513 and 0.9090 accuracy scores for CK+48 and KDEF test datasets, respectively. Meanwhile, Table II also shows that the steadiest models, in terms of accuracy standard deviation, are vanilla ViT and Swin for CK+48 and KDEF datasets, respectively. Recall, precision and F1 are also near to accuracy and stable. CrossViT model reaches its the best scores with least epoch compared to other models. Swin Transformer and vanilla ViT algorithms need

the most training epochs for CK+48 and KDEF datasets, respectively. Average scores of five experiments also were displayed in Figure 8a and 8b for CK+48 and KDEF datasets, respectively. Figure 8a and 8b reveals that Swin Transformer architecture exhibit the worst accuracy scores compared to other models. It is also possible to see most unstable models, in terms of accuracy standard deviation, are Swin Transformer and PiT for CK+48 and KDEF datasets, respectively. train loss and accuracy plots at every epoch were shared in in Figure 9 and Figure 10 for CK+48 and KDEF datasets, respectively. Figure 9a and 10a shows that CrossViT achieves the best train accuracy score and least train loss. However, it does not outperform other all models for test datasets.

Model selection in deep learning plays a significant role for many tasks including FER. It has a lot of effects on results and progress such as inference time, training time, test accuracy, and so on. Using complex models may increase the accuracy performance. However, complex models need to expensive hardware requirements and they have more inference time that hinder near real-time processing. Small models may run on low-level hardware. However, generally they are less accurate than complex models. Model selection has direct effect on budget, accuracy performance, inference time, and so on. All possible models should be evaluated to find optimal model to solve aimed tasks. Table I and II shows that although CrossViT has more parameter than DeiT, it is not able to outperform DeiT model. Moreover, Figure 9 and 10 reveal that CrossViT algorithm has lack of generalization capability since it learns train data so fast with less loss values but it is not able to exhibit same performance on test dataset. Although Swin Transformer has most parameter size, it achieves less performance than most of other models for both datasets. It

means using complex model is not solution at every time. Moreover, Swin Transformer has more inference time since it has hierarchical feature extraction and concatenation layers. PiT model, has convolution layers to extract embeddings, outperforms vanilla ViT that means using convolution layers to extract embeddings from the image helps to learn more useful features. Since convolution layers tends to extract local low-level features in its early layers [44], PiT combines local low-level features with ViT's global embedding extraction mechanism. All individual models have near accuracy, precision, recall and F1. Literature results for both CK+48 and KDEF datasets were shared in Table III. First column indicates the study, second column shows which dataset was used and the last column displays accuracy score of the studies. Although some studies perform better and worse compared to used Transformer based algorithms, It is seen that Transformer-based algorithms exhibit similar accuracy score compared to the literature.

In this paper, it is aimed to compare Transformer-based different image classification models for FER task. The five models that have different architectures but almost same trainable parameter sizes were selected. CK+48 and KDEF datasets were used to evaluate FER performance of proposed approaches. Experiment results show that although models have roughly same learnable parameter numbers, model performances and data processing time differ each other. It reveals that model architectures play an important role in performance in terms of accuracy and image processing capability. CNN and Transformer based hybrid PiT model outperforms all other models for both CK+48 and KDEF datasets in terms of accuracy. Moreover, PiT model has most image processing capability in one second. That makes PiT model suitable to use in FER applications. On the other hand, Swin transformer has worst image processing capability in one second. Swin Transformer and ViT model exhibit worst accuracy score for CK+48 and KDEF datasets, respectively. In further studies, domain specific datasets can be used for pretraining stage before transfer learning to increase performance of models. It is also possible to measure and compare accuracy and running time performance of Transformer and CNN based models.

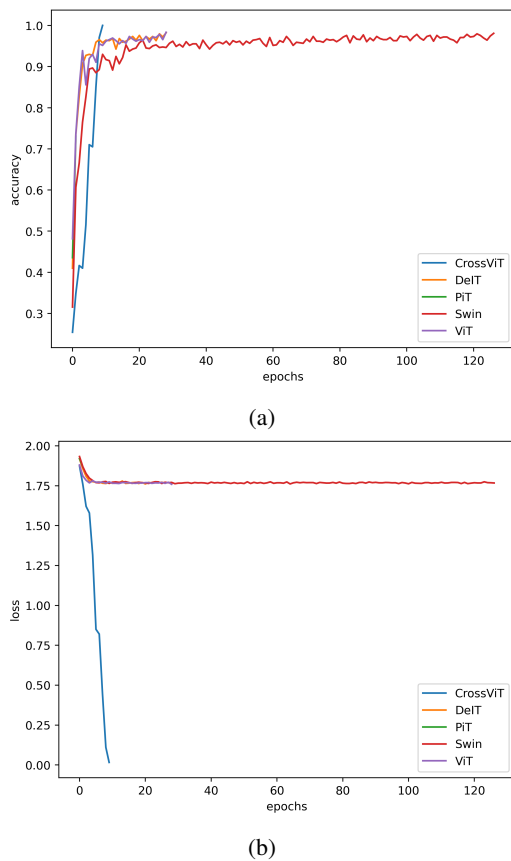


Fig. 9: (a) and (b) represent train accuracy and loss plots with respect to trained epoch for CK+48 dataset.

V. CONCLUSION

Facial Expression Recognition is an important study field in the literature. Human face mimics and their emotions are widely used such as human-computer interaction. To determine best models in terms of accuracy and running speed is important to develop suitable applications.

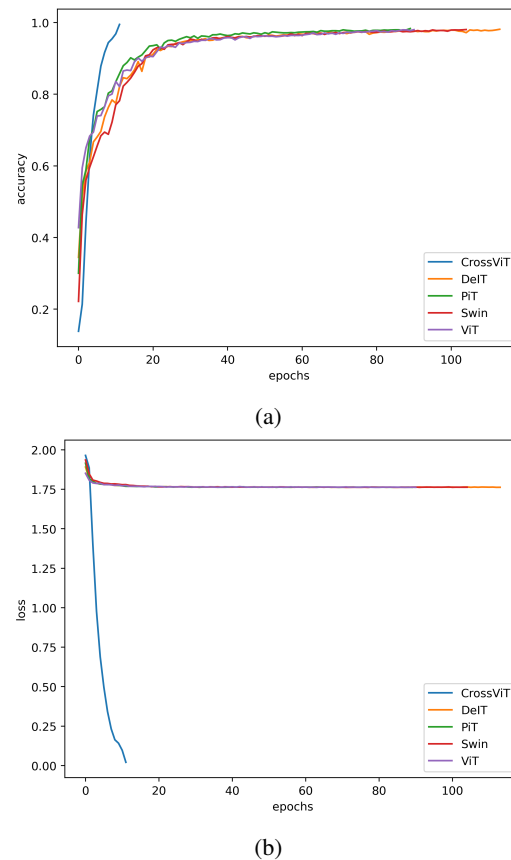


Fig. 10: (a) and (b) represent train accuracy and loss plots with respect to trained epoch for KDEF dataset.

REFERENCES

- [1] P. Ekman, "Facial expression and emotion." *American Psychologist*, vol. 48, no. 4, pp. 384–392, 1993. [Online]. Available: <https://doi.org/doi/10.1037/0003-066X.48.4.384>
- [2] L. E. Ishii, J. C. Nellis, K. D. Boahene, P. Byrne, and M. Ishii, "The importance and psychology of facial expression," *Otolaryngologic Clinics of North America*, vol. 51, no. 6, pp. 1011–1017, 2018-12. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S003066651830121X>
- [3] G. S. Shergill, A. Sarrafzadeh, O. Diegel, and A. Shekar, "Computerized sales assistants: the application of computer technology to measure consumer interest-a conceptual framework," 2008, publisher: California State University.
- [4] X.-Y. Tang, W.-Y. Peng, S.-R. Liu, and J.-W. Xiong, "Classroom teaching evaluation based on facial expression recognition," in *Proceedings of the 2020 9th International Conference on Educational and Information Technology*, ser. ICEIT 2020. Association for Computing Machinery, 2020-04-23, pp. 62–67. [Online]. Available: <https://doi.org/10.1145/3383923.3383949>
- [5] M. Sajjad, M. Nasir, F. U. M. Ullah, K. Muhammad, A. K. Sangaiah, and S. W. Baik, "Rasberry pi assisted facial expression recognition framework for smart security in law-enforcement services," *Information Sciences*, vol. 479, pp. 416–431, 2019-04. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0020025518305425>
- [6] G. Fu, Y. Yu, J. Ye, Y. Zheng, W. Li, N. Cui, and Q. Wang, "A method for diagnosing depression: Facial expression mimicry is evaluated by facial expression recognition," *Journal of Affective Disorders*, vol. 323, pp. 809–818, 2023-02. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S016503272201388X>
- [7] P. Ekman and W. V. Friesen, "Constants across cultures in the face and emotion," *Journal of Personality and Social Psychology*, vol. 17, pp. 124–129, 1971, place: US Publisher: American Psychological Association.
- [8] N. A. Sheth and M. M. Goyani, "A comprehensive study of geometric and appearance based facial expression recognition methods," *Int J Sci Res Sci Eng Technol*, vol. 4, no. 2, pp. 163–175, 2018-01-20. [Online]. Available: <https://ijsrset.com/IJSRSET184229>
- [9] T. Gwyn, K. Roy, and M. Atay, "Face recognition using popular deep net architectures: A brief comparative study," *Future Internet*, vol. 13, no. 7, p. 164, 2021.
- [10] A. Saeed, A. Al-Hamadi, R. Niese, and M. Elzobi, "Frame-based facial expression recognition using geometrical features," *Adv. in Hum.-Comp. Int.*, vol. 2014, p. 4-4, 2014-01-01. [Online]. Available: <https://doi.org/10.1155/2014/408953>
- [11] J.-H. Kim, B.-G. Kim, P. P. Roy, and D.-M. Jeong, "Efficient facial expression recognition algorithm based on hierarchical deep neural network structure," *IEEE Access*, vol. 7, pp. 41 273–41 285, 2019, conference Name: IEEE Access.
- [12] A. Barman and P. Dutta, "Facial expression recognition using distance and shape signature features," *Pattern Recognition Letters*, vol. 145, pp. 254–261, 2021-05. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0167865517302246>
- [13] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.
- [14] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," 2023-08-01. [Online]. Available: <http://arxiv.org/abs/1706.03762>
- [15] P. K. A. Vasu, J. Gabriel, J. Zhu, O. Tuzel, and A. Ranjan, "FastViT: A fast hybrid vision transformer using structural reparameterization," 2023-08-17. [Online]. Available: <http://arxiv.org/abs/2303.14189>
- [16] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 10012–10022.
- [17] C.-F. Chen, Q. Fan, and R. Panda, "CrossViT: Cross-attention multi-scale vision transformer for image classification," 2021-08-22. [Online]. Available: <http://arxiv.org/abs/2103.14899>
- [18] B. Heo, S. Yun, D. Han, S. Chun, J. Choe, and S. J. Oh, "Rethinking spatial dimensions of vision transformers," 2021-08-17. [Online]. Available: <http://arxiv.org/abs/2103.16302>
- [19] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, "Training data-efficient image transformers & distillation through attention," 2021-01-15. [Online]. Available: <http://arxiv.org/abs/2012.12877>
- [20] M. Rahul, N. Kohli, R. Agarwal, and S. Mishra, "Facial expression recognition using geometric features and modified hidden markov model," *International Journal of Grid and Utility Computing*, vol. 10, no. 5, pp. 488–496, 2019-01, publisher: Inderscience Publishers. [Online]. Available: <https://www.inderscienceonline.com/doi/abs/10.1504/IJGUC.2019.102018>
- [21] H. Chouhayebi, J. Riffi, M. A. Mahraz, A. Yahyaouy, H. Tairi, and N. Alioua, "Facial expression recognition based on geometric features," in *2020 International Conference on Intelligent Systems and Computer Vision (ISCV)*, 2020-06, pp. 1–6.
- [22] G. Sharma, L. Singh, and S. Gautam, "Automatic facial expression recognition using combined geometric features," *3D Res*, vol. 10, no. 2, p. 14, 2019-04-01. [Online]. Available: <https://doi.org/10.1007/s13319-019-0224-0>
- [23] D. A. Ibrahim, D. A. Zebari, F. Y. H. Ahmed, and D. Q. Zeebaree, "Facial expression recognition using aggregated handcrafted descriptors based appearance method," in *2021 IEEE 11th International Conference on System Engineering and Technology (ICSET)*, 2021-11, pp. 177–182, ISSN: 2470-640X.
- [24] H. Kaya, F. Gürpınar, S. Afshar, and A. A. Salah, "Contrasting and combining least squares based learners for emotion recognition in the wild," in *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*. ACM, 2015-11-09, pp. 459–466. [Online]. Available: <https://dl.acm.org/doi/10.1145/2818346.2830588>
- [25] D. Liu, X. Ouyang, S. Xu, P. Zhou, K. He, and S. Wen, "SAANet: Siamese action-units attention network for improving dynamic facial expression recognition," *Neurocomputing*, vol. 413, pp. 145–157, 2020-11-06. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S092523122031050X>
- [26] X. Pan, G. Ying, G. Chen, H. Li, and W. Li, "A deep spatial and temporal aggregation framework for video-based facial expression recognition," *IEEE Access*, vol. 7, pp. 48 807–48 815, 2019. [Online]. Available: <https://ieeexplore.ieee.org/document/8674456/>
- [27] M. Z. Uddin, W. Khaksar, and J. Torresen, "Facial expression recognition using salient features and convolutional neural network," *IEEE Access*, vol. 5, pp. 26 146–26 161, 2017. [Online]. Available: <http://ieeexplore.ieee.org/document/8119492/>
- [28] S. Minaee, M. Minaei, and A. Abdolrashidi, "Deep-emotion: Facial expression recognition using attentional convolutional network," *Sensors*, vol. 21, no. 9, p. 3046, 2021-04-27. [Online]. Available: <https://www.mdpi.com/1424-8220/21/9/3046>
- [29] M. G. Calvo and D. Lundqvist, "Facial expressions of emotion (KDEF): Identification under different display-duration conditions," *Behav Res*, vol. 40, no. 1, pp. 109–115, 2008-02-01. [Online]. Available: <https://doi.org/10.3758/BRM.40.1.109>
- [30] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews, "The extended cohn-kanade dataset (CK+): A complete dataset for action unit and emotion-specified expression," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops*. IEEE, 2010-06, pp. 94–101. [Online]. Available: <http://ieeexplore.ieee.org/document/5543262/>
- [31] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "PyTorch: An imperative style, high-performance deep learning library," 2019-12-03. [Online]. Available: <http://arxiv.org/abs/1912.01703>
- [32] L. Wang, Z. He, B. Meng, K. Liu, Q. Dou, and X. Yang, "Two-pathway attention network for real-time facial expression recognition," *Journal of Real-Time Image Processing*, vol. 18, no. 4, pp. 1173–1182, 2021.
- [33] S. Subudhiray, H. K. Palo, and N. Das, "Effective recognition of facial emotions using dual transfer learned feature vectors and support vector machine," *International Journal of Information Technology*, vol. 15, no. 1, pp. 301–313, 2023.
- [34] J. X. Yu, K. M. Lim, and C. P. Lee, "Move-cnns: Model averaging ensemble of convolutional neural networks for facial expression recognition," *IAENG International Journal of Computer Science*, vol. 48, no. 3, 2021.
- [35] Q. Hu, C. Wu, J. Chi, X. Yu, and H. Wang, "Multi-level feature fusion facial expression recognition network," in *2020 Chinese Control And Decision Conference (CCDC)*. IEEE, 2020, pp. 5267–5272.
- [36] K. Mohan, A. Seal, O. Krejcar, and A. Yazidi, "Fer-net: facial expression recognition using deep neural net," *Neural Computing and Applications*, vol. 33, no. 15, pp. 9125–9136, 2021.
- [37] N. Kumar HN, A. S. Kumar, G. Prasad MS, and M. A. Shah, "Automatic facial expression recognition combining texture and shape features from

prominent facial regions,” *IET Image Processing*, vol. 17, no. 4, pp. 1111–1125, 2023.

- [38] M. Kas, Y. Ruichek, R. Messoussi *et al.*, “New framework for person-independent facial expression recognition combining textural and shape analysis through new feature extraction approach,” *Information Sciences*, vol. 549, pp. 200–220, 2021.
- [39] S. Eng, H. Ali, A. Cheah, and Y. Chong, “Facial expression recognition in jaffe and kdef datasets using histogram of oriented gradients and support vector machine,” in *IOP Conference series: materials science and engineering*, vol. 705, no. 1. IOP Publishing, 2019, p. 012031.
- [40] R. V. Puthanidam and T.-S. Moh, “A hybrid approach for facial expression recognition,” in *Proceedings of the 12th International Conference on Ubiquitous Information Management and Communication*, 2018, pp. 1–8.
- [41] A. J. Obaid and H. K. Alrammahi, “An intelligent facial expression recognition system using a hybrid deep convolutional neural network for multimedia applications,” *Applied Sciences*, vol. 13, no. 21, p. 12049, 2023.
- [42] Y. Yaddaden, M. Adda, and A. Bouzouane, “Facial expression recognition using locally linear embedding with lbp and hog descriptors,” in *2020 2nd International Workshop on Human-Centric Smart Environments for Health and Well-being (IHSH)*. IEEE, 2021, pp. 221–226.
- [43] S. Barra, S. Hossain, C. Pero, and S. Umer, “A facial expression recognition approach for social iot frameworks,” *Big Data Research*, vol. 30, p. 100353, 2022.
- [44] M. D. Zeiler and R. Fergus, “Visualizing and understanding convolutional networks,” in *Computer Vision—ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6–12, 2014, Proceedings, Part 1 13*. Springer, 2014, pp. 818–833.



Abdülkadir Albayrak completed his bachelor’s, master’s, and doctoral studies in the field of computer engineering and developed algorithms based on traditional and deep learning methods for the classification of biomedical images and the detection of cellular structures (cancerous or normal cells) within these images for his theses. He has published many research articles in SCI-indexed journals, and has presented papers at various conferences. He achieved 2nd place worldwide in the competition titled “mitosis detection in histopathological images” organized by ICPR (International Conference Pattern Recognition). He is currently postdoctoral research fellow at the Department of Laboratory Medicine and Pathology.



Muhammed Cihad Arslanoğlu graduated from the Department of Electrical and Electronics Engineering at Dicle University for his undergraduate studies and is currently continuing his graduate studies in the same department. He has been involved in programming since high school and during his undergraduate studies, he developed numerous applications in image processing, artificial intelligence, and web technologies. Additionally, he works full-time as an Artificial Intelligence and Computer Vision specialist at Dicle Electricity Distribution Company.”



Hüseyin Acar received the B.S. degree in Electronics Engineering from Uludağ University, Bursa, Turkey, in 2006 and M.S. degree in Electrical-Electronics Engineering from Dicle University, Diyarbakır, Turkey, in 2010. He received the Ph.D. degree in Electrical-Electronics Engineering from Dicle University, Diyarbakır, Turkey, in 2020. He is currently an assistant professor at Dicle University Electrical-Electronics Engineering. His research interest includes machine learning, image processing, remote sensing, and embedded systems.