

# Dual-threshold Based Local Patch Construction Method for Manifold Approximation And Its Application to Facial Expression Analysis

1<sup>st</sup> S L Happy  
INRIA Sophia Antipolis  
France  
s-l.happy@inria.fr

2<sup>nd</sup> Antitza Dantcheva  
INRIA Sophia Antipolis  
France  
antitza.dantcheva@inria.fr

3<sup>rd</sup> Aurobinda Routray  
Department of Electrical Engineering  
Indian Institute of Technology Kharagpur, India  
aroutray@iitkgp.ac.in

**Abstract**—In this paper, we propose a manifold based facial expression recognition framework which utilizes the intrinsic structure of the data distribution to accurately classify the expression categories. Specifically, we model the expressive faces as the points on linear subspaces embedded in a Grassmannian manifold, also called as *expression manifold*. We propose the *dual-threshold based local patch (DTLP)* extraction method for constructing the local subspaces, which in turn approximates the expression manifold. Further, we use the affinity of the face points from the subspaces for classifying them into different expression classes. Our method is evaluated on four publicly available databases with two well known feature extraction techniques. It is evident from the results that the proposed method efficiently models the expression manifold and improves the recognition accuracy in spite of the simplicity of the facial representatives.

**Index Terms**—Facial expression analysis, manifold approximation, point to subspace distance.

## I. INTRODUCTION

Facial expression is the primary cue to recognize and interpret human emotion. With wide range of applications in human-computer interaction, biometrics, surveillance, retail, and health sector, facial expression analysis has become an active area of research in the recent years [1]. Paul Ekman broadly analyzed this channel from psychological point of view and postulated the universality of seven facial expressions, viz., anger, disgust, contempt, fear, happiness, sadness and surprise [2]. A growing literature has addressed the classification of the facial expressions into these discrete categories [1], [3].

The key to accurate expression recognition is the discriminating ability of the feature representation by capturing the subtle facial variations for different expressions. Most of the hand-crafted features used in literature [1] are based on facial geometry, appearance, motion, and deformation during different expressions. Emerging deep learning based approaches have significantly advanced the research and the related performances in expression classification [4]–[8]. Deep learning methods learn a feature representation that discriminates the expression classes efficiently. However, deep learning algorithms require huge training data for learning model weights. Lack of sufficient samples creates void in the feature space which affects the model performance. In such scenarios,

manifold based approaches are efficient at modelling the data distribution. This work mainly presents the *analysis of facial expressions* on the *expression manifold*. We use suitable face representations of existing works and propose an *improved subspace representation technique for expression manifold approximation*, thus improving the expression recognition performance.

Henceforth, we refer the facial feature vector  $\mathbf{f} \in \mathbb{R}^d$  as a ‘*face point*’ in  $d$ -dimensional space. The expression classification in the manifold of expressions was proposed in [9], where the facial deformation was embedded in a low-dimensional space using non-linear dimensionality reduction techniques. Usually, the extracted facial feature vector has very high dimensionality to encode the complex inter-personal and intra-personal facial attributes and deformations. However, dimensionality of these points can be reduced drastically so that it forms a smooth manifold with its intrinsic dimensionality. Thus, the basic assumption in this work is that the face points lie close on a low-dimensional manifold for the samples of a particular class. In other words, considering the facial images as points on the facial manifold, similar expressions lie in the local neighborhood of manifold of expressions.

The face points usually spread across a non-linear manifold, which is difficult to characterize using linear models. Nonlinear manifold based models have been investigated in the literature for its ability to tackle variation in face pose, expression intensity, blended expressions, inter-personal and intra-personal fluctuation in expression display. Jiying *et al.* [10] proposed a discriminant manifold non-negative matrix factorization algorithm to exploit the spatial neighborhood structure with constraints based on the manifold learning and graph embedding theory. Yuce *et al.* [11] utilized the multi-label Discriminant Laplacian Embedding (DLE) method to investigate the multi-label action unit detection problem by embedding the data on low dimensional manifolds while preserving multi-label correlation. Expressionlets were proposed in [12] for dynamic expression recognition by modelling the temporal and spatial information from video data samples.

Mohammadzade *et al.* [13] defined *expression subspace* as the subspace spanned by the face points from the same expression. They used PCA on the face points of an expression

class to represent an expression subspace. Thus, each expression class is represented by a subspace and the collection of all expression subspaces is assumed to lie in a Grassmann manifold [14]. Inspired by this work, this paper constructs expression subspaces for facial expression classification in expression manifold. However, we consider the local linear patches in the expression manifold to model it as a collection of linear subspaces. Thus, different to [13], our work constructs multiple subspaces for each expression class, thereby exploiting the non-linearity of the manifold in a better manner.

Though there exists some variation in the inter-individual facial deformation patterns associated with a certain expression, this variation is certainly less compared to the variation observed with multiple expression classes. Liu *et al.* [15] proposed prototypes-based facial expression recognition, which constructs statistical models for each facial regions for every expression class. The classification was carried out based on the similarity of the facial patterns to the learned prototypes. Similar to this work, we use the affinity score of the facial patterns to different linear subspaces as the feature vector for the classification task.

We propose an *efficient local subspace construction technique* followed by *affine-based classification* to improve the expression recognition accuracy. The proposed method extracts facial features and classifies them into different expression categories on expression manifold. First, the face points are clustered followed by linear subspace representation of the cluster points. The key contribution of the paper is the novel expression subspace construction scheme. We propose the *dual-threshold based local patch (DTLP)* construction method to locate the face points on local patches of the Grassmann manifold properly, thus improving the accuracy of expression recognition. The expression class for a sample is decided based on the affinity of the face point to different subspaces, which is discussed in section II-C. The performance of the proposed method is compared with other literature and evaluated extensively.

## II. PROPOSED METHOD

This section presents the proposed method in three stages: facial feature extraction, construction of expression subspaces, and affinity-based classification of facial patterns.

### A. Facial feature extraction

We used histogram of oriented gradient (HOG) and local binary pattern (LBP) features extracted from the face region as the face representative ( $\mathbf{f}$ ). HOG [16], being a gradient-based representation, encodes the shape and appearance of the objects in an image. The face image was first divided into several blocks; the histograms of the edge directions in those blocks were extracted and normalized; and finally the block features were concatenated to produce a single vector as the face representative or the face point. Similarly, the normalized LBP histograms from different blocks [17] are concatenated into a single feature histogram as the LBP representation.

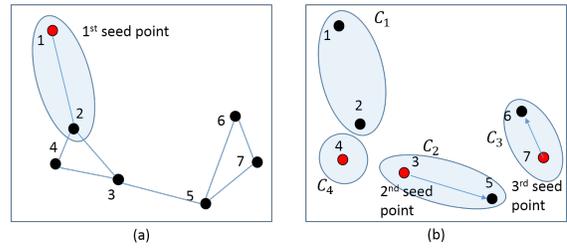


Fig. 1. Local subspace construction used by MLP. (a) Construction of  $C_1$ . The pair of points connected by solid lines implies they satisfy  $\Gamma(\cdot, \cdot) < \tau_1$ . Points marked in red are the seed points. (b) The final clustering result.

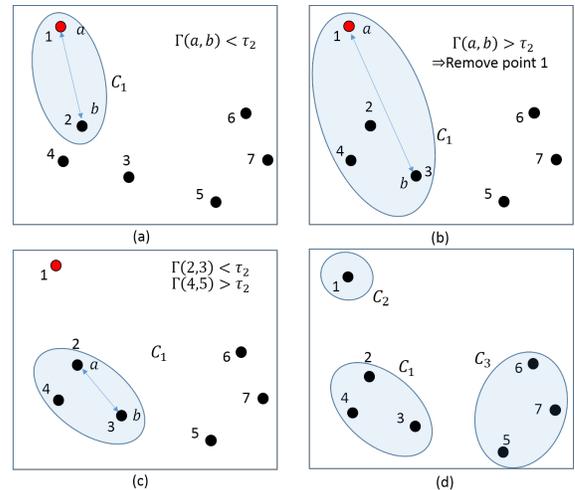


Fig. 2. The proposed (DTLP) local subspace construction method. The steps for  $C_1$  construction is shown in (a), (b) and (c). The final clustering result is shown in (d).

### B. Construction of expression subspaces

The non-linear expression manifold is approximated by a number of locally constructed subspaces. The face points  $\mathbf{f} \in \mathbb{R}^d$  lie on a higher dimensional manifold. Since the distance metrics become meaningless with increasing dimensions [18], finding close points in  $\mathbb{R}^d$  is a difficult task. The previous works [19] [20], to construct local linear models, typically use k-means clustering or hierarchical clustering techniques. However, such methods do not guarantee the linearity property. Moreover, the number of clusters needs to be specified in these methods. Wang *et al.* [21] propose maximal linear patch (MLP) to overcome these limitations. With the help of Euclidean and geodesic distances, they propose to grow the cluster from a seed point until the linearity constraint is broken. Similar to k-means clustering, this method also suffers from initial seed point selection. If the seed points are selected close to the actual cluster center, then the approximation is good. However, the problem comes when an end-member is selected as the seed point. This is illustrated in Fig. 1 (discussed later). This paper proposes a method to overcome this limitation by using dual-thresholds to add and remove points to a cluster. In spite of random seed point selection, the algorithm iteratively adds or removes the points from the

cluster and converges with points very close to each other as per the linearity constraint.

Let  $F_e = \{\mathbf{f}_{e,1}, \mathbf{f}_{e,2}, \dots, \mathbf{f}_{e,N_e}\}; \mathbf{f}_{e,n} \in \mathbb{R}^d$  be the  $N_e$  number of face points for the  $e$ th expression. Assuming these points lie on a non-linear expression manifold, we approximated each manifold by a number of subspaces  $S_{e,1}, S_{e,2}, \dots, S_{e,c_e}$ . Note that the number of subspaces ( $c_e$ ) varies from class to class. First, we partitioned the data  $F_e$  to different clusters  $C_{e,i}, i = 1, \dots, c_e$  such that the cluster points satisfy the linearity constraints, and

$$F_e = \bigcup_{i=1}^{c_e} C_{e,i} \quad (1)$$

$$\text{with } C_{e,i} \cap C_{e,j} = \phi \quad \text{for } i \neq j$$

where  $C_{e,i} = \{\mathbf{x}_1^{(i)}, \dots, \mathbf{x}_{n_{e,i}}^{(i)}\}$  contains the face points ( $\mathbf{x}_n^{(i)}$ ) of the cluster and  $n_{e,i}$  is the number of points in  $i$ th cluster. We used principal component analysis (PCA) to construct the local subspaces. Thus, the subspace ( $S_{e,i}$ ) is constructed from the points of cluster  $C_{e,i}$  and is represented by the sample mean ( $\mu_{e,i}$ ) and eigen direction matrix ( $\mathbf{P}_{e,i} \in \mathbb{R}^{d \times k_{e,i}}$ ) of the cluster points. The number of eigen vectors ( $k_{e,i}$ ) is set to preserve 95% of data variance.

The dual-threshold based local patch (DTLP) extraction method is proposed for construction of efficient local subspaces. We calculated the Euclidean ( $D_E$ ) and geodesic ( $D_G$ ) distances between each pair of training data points. We used the graph-based approach of Isomap constructed with five nearest neighborhood points to compute the geodesic distance between face points [22]. The ratio of the two distance measures between a pair of points  $\mathbf{a}$  and  $\mathbf{b}$  is given by,

$$\Gamma(\mathbf{a}, \mathbf{b}) = \frac{D_G(\mathbf{a}, \mathbf{b})}{D_E(\mathbf{a}, \mathbf{b})} \quad (2)$$

We used the geodesic distance ( $D_G$ ), the distance ratio ( $\Gamma$ ) and two threshold values ( $\tau_1, \tau_2$ ) to find out the points residing in the local neighborhood. First, we selected the cluster seed point randomly. We added new points to the cluster based on threshold  $\tau_1$ , which controls the compactness of the data points. Threshold  $\tau_2$  controls the degree of linearity. Higher the value of  $\tau_2$ , larger is the linearity deviation and vice versa. Note that  $\tau_2 > \tau_1$ . After adding new points to the cluster, we inspected the distance between the two farthest points  $\mathbf{a}$  and  $\mathbf{b}$ . If the linearity constraint is broken ( $\Gamma(\mathbf{a}, \mathbf{b}) > \tau_2$ ), we removed one of the points based on their closeness to the rest of the points in the cluster. This addition and removal process continues until no points can further be added to the current cluster. The pseudocode of the proposed method is provided in Algorithm 1.

MLP algorithm adds new points ( $\mathbf{x}_t$ ) to the cluster based on  $\Gamma(\mathbf{x}_i, \mathbf{x}_t) < \tau_1; \forall \mathbf{x}_i \in C$  constraint. As can be seen in Fig. 1, when  $C_1 = \{\mathbf{1}, \mathbf{2}\}$ , no more points (neither  $\mathbf{3}$  nor  $\mathbf{4}$ ) can be added to the cluster as it breaks the linearity constraint. Assuming the point  $\mathbf{3}$  to be the 2nd seed, it can add either  $\mathbf{4}$  or  $\mathbf{5}$  to the cluster. If  $\mathbf{5}$  is added, then no other points can be added to the cluster. Thus, it may end up constructing subspaces with clusters as shown in Fig. 1(b). On the contrary, DTLP first adds all the neighboring points to the cluster and then decides

which one should be removed if the linearity constraint is not satisfied. Cluster  $C_1$  starts with  $\mathbf{1}$  and adds point  $\mathbf{2}$ . Then, it simultaneously adds  $\mathbf{3}$  and  $\mathbf{4}$  (Fig. 2(b)) which in turn brakes the linearity constraint. Therefore, the point  $\mathbf{1}$  is removed as  $\mathbf{3}$  is closer to the other cluster points ( $\{\mathbf{2}, \mathbf{4}\}$ ). Thus, in spite of an end point being the seed, it may be removed afterward (see Fig. 2(c)). One can argue that point  $\mathbf{5}$  may be added to the cluster in the next iteration. In that case, point  $\mathbf{4}$  and  $\mathbf{5}$  will become the farthest points and if the linearity constraint is broken, then point  $\mathbf{5}$  will be again removed from the cluster. The final clustering results, as shown in Fig. 2(d), seems to be more appropriate for subspace construction.

### C. Affinity-based classification

We used the distance of the test face point from all the subspaces as a measure of similarity to classify it into different

---

**Algorithm 1** Proposed manifold approximation method: Local subspace construction using DTLP

---

**Input:** Training data points  $F_e = \{\mathbf{f}_{e,1}, \mathbf{f}_{e,2}, \dots, \mathbf{f}_{e,N_e}\}$

**Output:** Data clusters  $C_{e,i} = \{\mathbf{x}_1^{(i)}, \dots, \mathbf{x}_{n_{e,i}}^{(i)}\}; i = 1, \dots, c_e$

- 1: Initialization:  $i \leftarrow 0; X \leftarrow F_e$  with  $\mathbf{x}_n = \mathbf{f}_{e,n}$
  - 2: Find the pair-wise Euclidean ( $D_E$ ) and geodesic ( $D_G$ ) distance matrices. Compute  $\Gamma$  for each pair using (2).
  - 3: **while**  $X$  is not empty **do**
  - 4:   Update:  $i \leftarrow i + 1; C_{e,i} = \phi$
  - 5:   Select a seed point ( $\mathbf{x}_1^{(i)} = \mathbf{x}_n = \mathbf{f}_{e,n}$ ) randomly from  $X$   
 $C_{e,i} \leftarrow C_{e,i} \cup \mathbf{x}_n$
  - 6:   Identify all the points  $\{\mathbf{x}_t\}_{t=1}^m$  from  $X$  as the neighborhood points of  $\mathbf{x}_n^{(i)} \in C_{e,i}$  if  $\mathbf{x}_t$  satisfies  
 $\Gamma(\mathbf{x}_t, \mathbf{x}_n^{(i)}) < \tau_1, \quad \text{for any } \mathbf{x}_n^{(i)} \in C_{e,i}$
  - 7:    $C_{e,i} \leftarrow C_{e,i} \cup \{\mathbf{x}_t\}_{t=1}^m$
  - 8:   Find the farthest two points  $\{\mathbf{x}_a^{(i)}, \mathbf{x}_b^{(i)}\}$  in the cluster based on  $\Gamma$ .  
 $\Gamma(\mathbf{x}_a^{(i)}, \mathbf{x}_b^{(i)}) \leftarrow \max(\Gamma(\mathbf{x}_p^{(i)}, \mathbf{x}_q^{(i)}); p, q = 1, \dots, n_{e,i})$
  - 9:   **if**  $\Gamma(\mathbf{x}_a^{(i)}, \mathbf{x}_b^{(i)}) > \tau_2$  **then**   ▷ (Remove one of the points  $\{\mathbf{x}_a^{(i)}, \mathbf{x}_b^{(i)}\}$  from the cluster  $C_{e,i}$  based on the geodesic distance from rest cluster points)
  - 10:     **if**  $\sum_n D_G(\mathbf{x}_a^{(i)}, \mathbf{x}_n^{(i)}) > \sum_n D_G(\mathbf{x}_b^{(i)}, \mathbf{x}_n^{(i)})$   
       where  $\mathbf{x}_n^{(i)} \in C_{e,i}; n \neq \{a, b\}$  **then**  
        $C_{e,i} \leftarrow C_{e,i} \setminus \{\mathbf{x}_a^{(i)}\}$
  - 11:     **else**  
        $C_{e,i} \leftarrow C_{e,i} \setminus \{\mathbf{x}_b^{(i)}\}$
  - 12:     **end if**
  - 13:     Go to step 8
  - 14:   **end if**
  - 15:   **if** no new points are added to  $C_{e,i}$  **then**  
        $C_{e,i}$  contains the required points.  
       ▷ Go for the next cluster.
  - 16:   **else**  
       Go to step 6, and add new points to  $C_{e,i}$ .
  - 17:   **end if**
  - 18:    $X \leftarrow X \setminus \{\bigcup_n \mathbf{x}_n^{(i)}\}$
  - 19: **end while**
-

classes. In [21], the point to subspace distance is defined as:

$$d(\mathbf{x}, S_{e,i}) = \min_{\mathbf{y} \in S_{e,i}} \|\mathbf{x} - \mathbf{y}\| \quad (3)$$

where  $\mathbf{y}$  is a point on the subspace  $S_{e,i}$  and  $\|\cdot\|$  is the  $L_2$  norm of the vector. Similar work is carried out in [23]. In fact,  $\mathbf{y}$  is the closest point of  $\mathbf{x}$  on  $S_{e,i}$ , which can be found by projecting  $\mathbf{x}$  on the subspace. Thus,

$$d(\mathbf{x}, S_{e,i}) = \|\mathbf{x} - \mu_{e,i} - \mathbf{P}_{e,i} \mathbf{P}_{e,i}^T (\mathbf{x} - \mu_{e,i})\| \quad (4)$$

where  $\mathbf{P}_{e,i}$  is principal component matrix which forms a set of orthonormal basis of the subspace. Both  $\mathbf{P}_{e,i}$  and  $\mu_{e,i}$  are obtained from PCA.

Given the training face points, we calculated their affinity from all the subspaces  $S_{e,i}; e = 1, \dots, 7$  for seven classes,  $i = 1, \dots, c_e$  and use them as the training vectors to construct linear support vector machine (SVM) classifiers. The test input face points are classified based on their distance from all the subspaces.

### III. EXPERIMENTS AND RESULTS

Four standard facial expression databases, viz. Cohn-Kanade (CK+) [24], JAFFE [25], Radboud faces Database (RaFD) [26], and FACES [27], were used in our experiments. All these databases contain samples from the seven universal expressions, except FACES which contains six expression classes including neutral. CK+ database consists of 327 image sequences, which starts from neutral and ends with a peak expression. We used the peak expressive image from each sequence. Similarly, 1407 frontal expressive faces of seven expressions from RaFD database, 213 samples from JAFFE, and 2052 samples from FACES dataset were used in our experiments.

After detecting the face, we cropped the face and resized it to  $96 \times 96$  resolution. For extracting HOG features, we divided the face into blocks of size  $8 \times 8$  and used 9 angular histogram bins for each block. Thus, we obtained a feature vector of dimension  $12 \times 12 \times 9 = 1296$ . Similarly, for LBP features, we first divided the face into  $8 \times 8$  grids and concatenated the LBP histograms from each region which produced a vector of length  $8 \times 8 \times 256 = 16384$ . We used linear SVM with one-vs-one scheme as the classifier. Ten-fold cross-validation was adopted to produce the results.

The effectiveness of the proposed local subspace construction approach is substantiated in Fig. 3. For a fair comparison, we kept  $\tau_1 = 1.6$  for the local model construction by MLP and  $\tau_1 = 1.1$  and  $\tau_2 = 1.6$  for DTLP. Fig. 3 shows the results obtained with these settings in all the four databases. In all cases, the proposed DTLP method performed better than the MLP method. Our method achieved an accuracy of 98.57% and 94.78% in RaFD and FACES respectively, which are close to the accuracy of state-of-art techniques. Moreover, these results were obtained without proper parameter tuning of the classifiers and the features, which can be further improved.

The performance of the proposed manifold based expression analysis depends on the degree of appropriate manifold approximation. The reduced performance in JAFFE and CK+

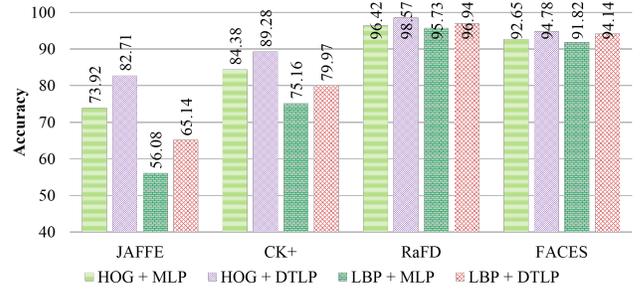


Fig. 3. Comparison of facial expression classification accuracy obtained in different databases with MLP and DTLP.

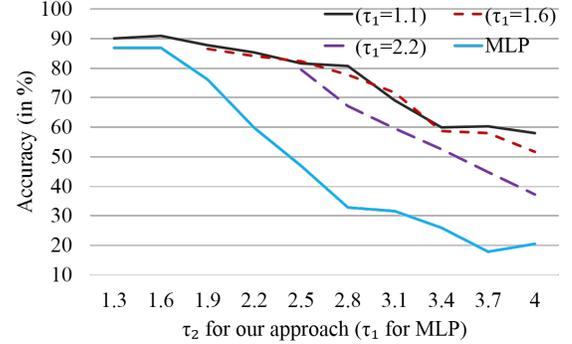


Fig. 4. Effect of parameter variation (varying  $\tau_1$  and  $\tau_2$ ) in DTLP and MLP on CK+ database using HOG features. Note that the horizontal axis represents  $\tau_1$  in case of MLP.

databases could be due to the relatively low sample size, which affects the creation of suitable manifold approximation. On the contrary, the classification performance was good for both RaFD and FACES, mainly due to the large sample size of these datasets.

Fig. 4 shows the effect of different parameter selection for DTLP on the classification accuracy. As can be observed, the accuracy decreased with increasing  $\tau_2$  as the local patches becomes nonlinear with the increasing value of  $\tau_2$ . Compared to the performance of MLP, the proposed method performed better for a similar set of parameters. With the increase in linearity deviation, the accuracy decreased more rapidly in MLP than the DTLP method.

The performance comparison of RaFD and FACES datasets with state-of-the-art methods are reported in Table I and II

TABLE I  
COMPARISON OF SEVEN CLASS CLASSIFICATION ACCURACY ON RAFD DATABASE.

| Methods                     | Validation settings | Accuracy     |
|-----------------------------|---------------------|--------------|
| Metric learning [28]        | 10 fold             | 95.95        |
| W-CR-AFM [5]                | train-test split    | 96.27        |
| BAE-BNN-3 [4]               | 5 fold              | 96.93        |
| TLCNN+FOS [8]               | 4 fold              | 97.75        |
| Carcagni <i>et al.</i> [29] | 10 fold             | 98.5         |
| <b>Proposed Method</b>      | 10 fold             | <b>98.57</b> |

TABLE II  
COMPARISON OF EXPRESSION RECOGNITION ACCURACY ON FACES  
DATABASE.

| Methods                | Accuracy     |
|------------------------|--------------|
| Guo <i>et al.</i> [30] | 84.68        |
| Joint-Learn [31]       | 92.19        |
| Wu <i>et al.</i> [32]  | 94.12        |
| <b>Proposed Method</b> | <b>94.78</b> |

respectively. The proposed manifold based expression analysis approach achieved the best recognition accuracy in both datasets. As can be observed in Table I, our method outperforms the performance of some deep learning models, indicating the significance of the manifold approximation through DTLP.

#### IV. CONCLUSIONS

In this paper, we propose a dual-threshold based local subspace construction method and applied it to approximate the expression manifold. The face points are further classified according to the affinity score of the point from all the subspaces. The proposed method is analyzed extensively with CK+, JAFFE, FACES and RaFD databases, which demonstrates the improvement in expression recognition accuracy with the proposed DTLP method. Since the classification performance depends on the discriminating features, we strongly believe that the use of sophisticated discriminative features will further improve the accuracy. In future, we plan to use the feature representation from the deep learning models with the proposed manifold approximation method. DTLP is a generic method which can also be applied to other applications, such as face recognition, object identification, texture classification, etc.

#### REFERENCES

- [1] E. Sariyanidi, H. Gunes, and A. Cavallaro, "Automatic analysis of facial affect: A survey of registration, representation, and recognition," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 37, no. 6, pp. 1113–1133, 2015.
- [2] P. Ekman, *Emotions revealed: Recognizing faces and feelings to improve communication and emotional life*. New York: Times Books, 2003.
- [3] S. Happy and A. Routray, "Recognizing subtle micro-facial expressions using fuzzy histogram of optical flow orientations and feature selection methods," in *Comput. Intell. for Pattern Recog.*, 2018, pp. 341–368.
- [4] W. Sun, H. Zhao, and Z. Jin, "An efficient unconstrained facial expression recognition algorithm based on stack binarized auto-encoders and binarized neural networks," *Neurocomputing*, vol. 267, pp. 385–395, 2017.
- [5] B.-F. Wu and C.-H. Lin, "Adaptive feature mapping for customizing deep learning based facial expression recognition model," *IEEE Access*, vol. 6, pp. 12 451–12 461, 2018.
- [6] Y. Liu, X. Yuan, X. Gong, Z. Xie, F. Fang, and Z. Luo, "Conditional convolution neural network enhanced random forest for facial expression recognition," *Pattern Recognition (PR)*, vol. 84, pp. 251–261, 2018.
- [7] S. Happy, A. Dantcheva, A. Das, R. Zeghari, P. Robert, and F. Bremond, "Characterizing the state of apathy with facial expression and motion analysis," in *IEEE Int. Conf. on Automatic Face & Gesture Recognition (FG 2019)*, 2019.
- [8] Y. Zhou and B. E. Shi, "Action unit selective feature maps in deep networks for facial expression recognition," in *Int. Joint Conf. on Neural Networks (IJCNN)*. IEEE, 2017, pp. 2031–2038.
- [9] Y. Chang, C. Hu, R. Feris, and M. Turk, "Manifold based analysis of facial expression," *Image and Vision Computing*, vol. 24, no. 6, pp. 605–614, 2006.
- [10] W. Jiying, X. Shenglin, Y. Yi, and L. Bo, "A discriminant manifold non-negative matrix factorization algorithm for facial expression recognition," in *IEEE Int. Conf. on Signal Processing*, 2016, pp. 1010–1015.
- [11] A. Yüce, H. Gao, and J.-P. Thiran, "Discriminant multi-label manifold embedding for facial action unit detection," in *Int. Conf. and Workshops on Automatic Face and Gesture Recognition (FG)*, vol. 6, 2015, pp. 1–6.
- [12] M. Liu, S. Shan, R. Wang, and X. Chen, "Learning expressionlets via universal manifold model for dynamic facial expression recognition," *IEEE Trans. on Image Processing*, vol. 25, no. 12, pp. 5920–5932, 2016.
- [13] H. Mohammadzade and D. Hatzinakos, "Projection into expression subspaces for face recognition from single sample per person," *IEEE Trans. on Affective Computing*, vol. 4, no. 1, pp. 69–82, Jan 2013.
- [14] J. Hamm and D. D. Lee, "Grassmann discriminant analysis: a unifying view on subspace-based learning," *Int. Conf. on Machine Learning*, pp. 376–383, 2008.
- [15] M. Liu, R. Wang, S. Shan, and X. Chen, "Learning prototypes and similes on grassmann manifold for spontaneous expression recognition," *Computer Vision and Image Understanding*, vol. 147, pp. 95–101, 2016.
- [16] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," *IEEE Conf. on Computer Vision and Pattern Recognition*, vol. 1, pp. 886–893, 2005.
- [17] S. Moore and R. Bowden, "Local binary patterns for multi-view facial expression recognition," *Computer Vision and Image Understanding*, vol. 115, no. 4, pp. 541–558, 2011.
- [18] C.-M. Hsu and M.-S. Chen, "On the design and applicability of distance functions in high-dimensional data space," *IEEE Trans. on Knowledge and Data Engineering*, vol. 21, no. 4, pp. 523–536, 2009.
- [19] W. Fan and D.-Y. Yeung, "Locally linear models on face appearance manifolds with application to dual-subspace based classification," *Computer Vision and Pattern Recognition*, vol. 2, pp. 1384–1390, 2006.
- [20] T.-K. Kim, O. Arandjelovic, and R. Cipolla, "Learning over sets using boosted manifold principal angles (bompa)," *British Machine Vision Conf.*, pp. 58–1, 2005.
- [21] R. Wang, S. Shan, X. Chen, and W. Gao, "Manifold-manifold distance with application to face recognition based on image set," *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 1–8, 2008.
- [22] J. B. Tenenbaum, V. De Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.
- [23] N. Bajaj, S. Happy, and A. Routray, "Dynamic model of facial expression recognition based on eigen-face approach," in *IEEE Proc. of Green Energy and Systems Conf.* IEEE, 2013.
- [24] 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," *IEEE Computer Vision and Pattern Recognition - Workshops on human communicative behavior analysis*, pp. 94–101, 2010.
- [25] M. Lyons, S. Akamatsu, M. Kamachi, and J. Gyoba, "Coding facial expressions with gabor wavelets," *IEEE Int. Conf. and Workshops on Automatic Face and Gesture Recognition (FG)*, pp. 200–205, 1998.
- [26] O. Langner, R. Dotsch, G. Bijlstra, D. H. Wigboldus, S. T. Hawk, and A. van Knippenberg, "Presentation and validation of the radboud faces database," *Cognition and Emotion*, vol. 24, no. 8, pp. 1377–1388, 2010.
- [27] N. C. Ebner, M. Riediger, and U. Lindenberger, "Facesa database of facial expressions in young, middle-aged, and older women and men: Development and validation," *Behavior research methods*, vol. 42, no. 1, pp. 351–362, 2010.
- [28] B. Jiang and K. Jia, "Robust facial expression recognition algorithm based on local metric learning," *Journal of Electronic Imaging*, vol. 25, no. 1, pp. 013 022–013 022, 2016.
- [29] P. Carcagni, M. Del Coco, M. Leo, and C. Distanto, "Facial expression recognition and histograms of oriented gradients: a comprehensive study," *SpringerPlus*, vol. 4, no. 1, p. 645, 2015.
- [30] G. Guo, R. Guo, and X. Li, "Facial expression recognition influenced by human aging," *IEEE Trans. on Affective Computing*, vol. 4, no. 3, pp. 291–298, 2013.
- [31] Z. Lou, F. Alnajjar, J. M. Alvarez, N. Hu, and T. Gevers, "Expression-invariant age estimation using structured learning," *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 40, no. 2, pp. 365–375, 2018.
- [32] S. Wu, S. Wang, and J. Wang, "Enhanced facial expression recognition by age," in *IEEE Int. Conf. and Workshops on Automatic Face and Gesture Recognition (FG)*, vol. 1. IEEE, 2015, pp. 1–6.