Contents lists available at Science-Gate

International Journal of Advanced and Applied Sciences

Journal homepage: http://www.science-gate.com/IJAAS.html

A novel selection model of random features for the estimation of facial expression

Do Nang Toan [1,*], Huynh Cao Tuan [2], Ha Manh Toan [3]

[1]The Information Technology Institute (ITI), Vietnam National University, Hanoi, Vietnam
[2]Lac Hong University, Dong Nai, Vietnam
[3]Institute of Information Technology, Vietnam Academy of Science and Technology, Hanoi, Vietnam

ARTICLE INFO

Article history:
Received 9 January 2018
Received in revised form
25 March 2018
Accepted 3 April 2018

Keywords:
Facial expression
Facial emotion
Active appearance models
Japanese female facial expression

ABSTRACT

Estimation of facial expressions has been an important focus in several practical applications of machine vision and virtual reality; such as assessing the satisfaction level of customers in using products/services or modelling virtual broadcasters. In this study, we propose a novel approach in estimating the facial expressions based on the automatic mechanism to randomly select facial geometric features and organize them into a tree model. By testing with the standard dataset JAFFE, it is found that our proposed model is efficient and effective and should be considered in the practical implementation.

1. Introduction

Facial expression has been one of the interesting research topics in several machine vision and virtual reality problems in practice. The researches can be classified into two major categories, including: (1) human facial expressions in combination with the face detection; and (2) human facial states in combination with face models. The recent advances in the computational hard-wares and related equipment offer huge advantages in developing human imitation models; especially, the representation of human facial expressions in 3D virtual reality has been widely employed in several applications, among which we can easily name the fictitious films called Avatar or Van Helsing with the monsters and wolf-men with their fine expressions and movements.

Besides, the identification or estimation of different states of facial expressions has also popularly used in other applications, such as the system on Google Glass eyeglasses to analyze human face developed by Fraunhofer IIS as shown in Fig. 1. In such applications, the quick and precise capture of facial features on the human face becomes one of the extremely important stages to produce satisfactory outputs.

Literally, the facial expressions are resulted from the movements of facial muscles which temporarily deform the facial features such as the eyelids, eyebrows, nose, lips, and skins (wrinkles, cutis anserine). In addition, the same facial expression can be differently interpreted as it heavily depends on the personal characteristics (age, gender, health, etc.). As such, there have been many different approaches for the facial identification and estimation problems. For example, by using facial features, some researchers use facial geometric points (Valstar and Pantic, 2007; Lucey et al., 2006), or image profile (Bartlett et al., 2006; Jiang et al., 2011), or both (Tian et al., 2001); whereas others consider the changes on the face chronologically (Jiang et al., 2011; Zhao and Pietikainen, 2007).

In this paper, we propose a novel approach to automatically select geometric facial features and organize them in a decision tree model to represent facial expressions.

2. Literature review

Active Appearance Models (AAM) is an algorithm used to determine key points on each face where each point has its own specific characteristics (Cootes et al., 2001; Viola & Jones, 2001). In AAM, a statistical model respective to the appearance of the object in an image combined with an optimal algorithm is used to identify the parameters representing the most appropriate model for the image. However, Baker and Matthews (2001) improved the performance of AAM by combining critical information obtained from 2D and 3D

* Corresponding Author.
Email Address: dntoan@vnu.edu.vn (D. N. Toan)
https://doi.org/10.21833/ijaas.2018.06.008

models; and they found that the improved approach results in better accuracy and real-time convergence in several particular cases (Xiao et al., 2004). Specifically, the interested object in an image is modeled with a set of control points describing its shape and structure which is actually the sample values of the image intensity within certain regions bordered by a set of control points as shown in Fig. 2 (our actual experiment).

Fig. 1: Google Glass developed by Fraunhofer IIS

Fig. 2: Shape and structure of image object in AAM

Literally, a statistical model for an object must be able to satisfactorily fully describe the variations of its shape and of its structure as well as the statistical correlation among them. The key controversial issues in this approach include the construction of a statistical model for the image object and the design of an optimal searching algorithm. Particularly, the construction of the model for the object consists of: (1) constructing a mathematical model for its shape and a model for the image structure; and (2) combining the two models to establish the expected model. And the optimal searching algorithm used in AAM is designed in such a way that the parameters of the model can be automatically estimated from the dataset and result in a constructed sample image which best describe the input image in term of minimizing the difference between the constructed and the input images.

Hien and Toan (2016) proposed a novel approach in statistically analyzing the shape parameters describing human face to detect nodding behaviors because there is a significant difference between a head in normal position and a nodding one. Fig. 3 shows the distribution of some shape parameters (Hien and Toan, 2016).

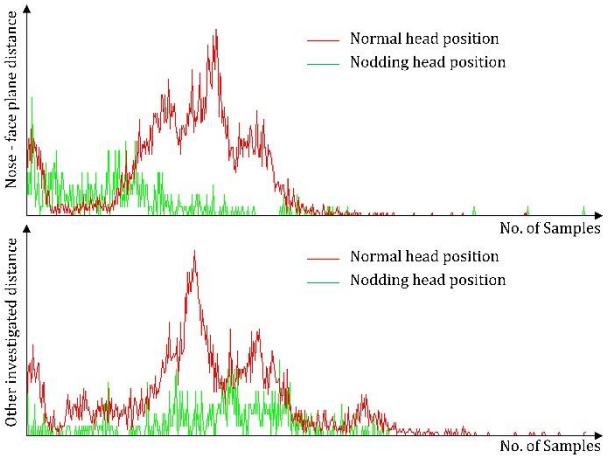

Fig. 3: Distribution of some shape parameters

From a set of input images labeled with control points and head position (normal or nodding), the model parameters were automatically and directly obtained from the data collected. From practical experiments with some parameters including point-point distance, point-edge distance, and triangle area, they conducted some statistics on the respective values and identified appropriate split thresholds. Some key characteristics with good split ability are used to detect nodding behavior.

3. Proposed algorithm

Literally, a shape human face can be effectively presented with a set of control points as discussed in Section 2; and the shape parameters are normally determined by the coordinates of one or some points in the set. As such, a shape parameter is actually the distance between 2 points or an area of any 3 points.

Consequently, we have many shape parameters to be considered. For example, for a set with 68 points, we consider the distance between 2 points; then we will have totally 2278 parameters. When the facial expression is changed, the coordinates of the points are also changed, resulting in the change of the parameters of the facial features. As a consequence, there is a significant change in the parameters which can be used to detect the change in the facial expressions.

Instead of manually identifying the geometric parameters as discussed in (Hien and Toan, 2016), we propose an automatic mechanism to select and organize them under a decision tree to estimate the facial expressions. Specifically, the tree consists of nodes; each node is a decision function learned for a particular facial feature. After a human face is successfully positioned, the set of control points is sent to the decision tree for critical analysis. For the final conclusion, the estimation values of facial expressions are determined by the average of all learning samples at the end-node.

3.1. A model of decision tree

Developed from previous researches by Hien and Toan (2016), our decision tree is established from

the set of training dataset with the following structure:

$$\{(I_s, v_s, w_s): s = 1,2, \dots, S\}$$

where, S is the size of the sample, $v_s \in [0,1]$ is the correct label value of sample I_s and w_s is respectively the weight of the sample.

In our study, w_s represents the importance of each input sample in the training dataset. At each node, we select the most appropriate function which provides the best classification ability for the dataset, i.e. the objective function obtains its minimum value. Particularly, our proposed objective function is determined by:

$$WMSE(I, v, w) = \sum_{(I,v,w)\in C_0} w(v - \bar{v}_0)^2 + \sum_{(I,v,w)\in C_1} w(v - \bar{v}_1)^2$$

where, C_0 and C_1 are two clusters in the training dataset, respectively the results of 0 and 1. The $\bar{v}_0$ and $\bar{v}_1$ are respectively the averages of the label values in C_0 and C_1.

In other words, at each node, we consider decision functions established from the shape features and identify the best one to minimize the objective function. Thus, from the original dataset, at each learning stage of each node during the construction of the tree, the training dataset are accordingly classified into two clusters. Our proposed algorithm for the learning of each node is as shown in Fig. 4.

```
Input: U ={(Is,vs,ws): s=1,2,...,S}
Output: T = {N0, N1,...,NS-1}
begin
T:= Ø
Idx0 = {0, 1, 2,...,S-1}
Stack:= Ø
push(Stack, {N0, Idx0});
while (Stack ≠ Ø)
    {Ni,Idxi}:= pop(Stack);
    if (Ni.level>=MAX_DEPTH)
        Continue;
    else
        min_err := MAX_VALUE;
        best_decidefunc := null;
        for each decidefunc
            e := WMSE(decidefunc, U, Idxi);
            if (e< minerr)
            best_decidefunc := decidefunc;
            min_err := e;
         endif
    endfor
    setupNode(Ni, best_decidefunc, U, Idxi);
    {Idxi*2+1, Idxi*2+2} := SplitDataSet(U, Idxi,
              best_decidefunc);
    push(Stack, {Ni*2+1, Idxi*2+1});
    push(Stack, {Ni*2+2, Idxi*2+2});
    endif
endwhile
end
```

Fig. 4: proposed algorithm for the learning of each node

3.2. Shape parameters

From the model of decision tree described above, shape parameters should be determined. We propose using the following three approaches.

3.2.1. Line_ Line

With 4 input points p_1, p_2, p_3, and p_4, let $d(p_i,p_j)$ denote the distance between point p_i and p_j. Then our Line_Line parameter is determined by:

$$f_{Line_{Line}}(p_1, p_2, p_3, p_4) = \frac{d(p_1,p_2)}{d(p_3,p_4)}.$$

3.2.2. Triangle_ Triangle

With 6 input points p_1, p_2, p_3, p_4, p_5, and p_6, let a, b, c denote the distances of the three points p_i, p_j, and p_k, i.e. $a = d(p_i,p_j)$, $b = d(p_j,p_k)$, $c = d(p_i,p_k)$. Then, let $S(p_i,p_j,p_k)$ denote an area of triangular formed by p_i, p_j, and p_k; thus, we have $S(p_i,p_j,p_k) = S(a,b,c)$ and $S(a,b,c)$ is determined by Heron's formula as the following:

$$S(a,b,c) = \frac{1}{4}\sqrt{(a+b+c)(a+b-c)(a-b+c)(b+c-a)}$$

Finally, our Triangle_Triangle parameter is determined by:

$$f_{Triangle-Triangle}(P_1, P_2, P_3, P_4, P_5, P_6) = \frac{S(P_1,P_2,P_3)}{S(P_4,P_5,P_6)}$$

3.2.3. LineLine_ LineLine

With 8 input points p_1, p_2, p_3, p_4, p_5, p_6, p_7, and p_8, let $d(p_i,p_j)$ denote the distance between point p_i and p_j. Then our LineLine_LineLine parameter is determined by:

$$f_{Line_{Line}}(p_1, p_2, p_3, p_4, p_5, p_6, p_7, p_8) = \frac{d(p_1,p_2)+d(p_3,p_4)}{d(p_5,p_6)+d(p_7,p_8)}.$$

3.3. Decision function

A decision function is constructed from a certain feature and a decision is made by comparing the obtained value against a threshold. The threshold is computed from an input dataset at each node. Specifically, in constructing a decision function at each node, a set of shape parameters are randomly generated, thus a set of decision functions are accordingly generated. With the threshold estimation for each function in the set, a function with the minimum error value is selected for the node. The algorithm for the threshold computation is proposed as shown in Fig. 5.

4. Empirical tests

Our empirical tests use the Japanese Female Facial Expression (JAFFE) database which contains 213 images of 7 facial expressions (6 basic facial expressions including happy, sad, surprise, angry, disappointed, panic and one neutral) posed by 10 Japanese female models. Each image has been rated on 6 emotion adjectives by 60 Japanese subjects. A 5-level scale was used for each of the 6 adjectives (5-high, 1-low). Specifically, the files contain semantic

rating data from psychological experiments using the images and the expression labels on the images represent the predominant expression in that image - the expression that the subject was asked to pose.

```
Input: U ={(Is,vs,ws): s=1,2,...,S}
Output: threshold
begin
    for each Is
        feats := calc_feat(Is);
    endfor
    sort(feats);
    best_threshold := 1;
    threshold := 1;
    min_error := WMSE(U);
    for i:=1 to S
        threshold := (feati-1+feati)/2
        error:= WMSE(U);
        if (error<min_error)
            min_error := error
            best_threshold:=threshold
        endif
    endfor
    threshold := best_threshold;
end
```

Fig. 5: Algorithm for algorithm for the threshold computation

JAFFE database has been preferably used in several problems relating to facial expression detection; for example, Bashan and Venayagamoorthy (2008) used Gabor filter in combination with learning vector quantization (LVQ) to extract expected features from the database. Their approach successfully obtains an accuracy level of 90.2%. Or, Oliveira et al. (2011) could successfully obtain an accuracy level of 94% by using 2PDCA and SVM to extract expected features.

In our empirical test with the JAFFE database, firstly, control points on all images in the database are positioned as shown in Fig. 6 so that shape parameters are calculated.

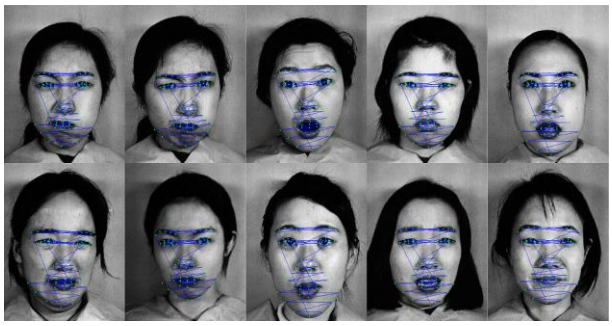

Fig. 6: Positioning of control points

With the control points identified on each image, we test the performance of our proposed approach through a cross validation technique; particularly, the dataset is divided into 6 groups for 6 different tests. In each test, one group is used for official test while other five groups are used for training. As such, the average of estimation error on each type of facial expression from our proposed approach is easily computed and shown in Table 1.

From our empirical tests, we have also found that a statistical relationship between thresholds and its accuracy as shown in Fig. 7.

Table 1: Average error on JAFFE database

No.	Facial Expression	Average Error
1	Happy	0.074626
2	Sad	0.048493
3	Surprise	0.086469
4	Angry	0.059132
5	Disappointed	0.079005
6	Panic	0.038198

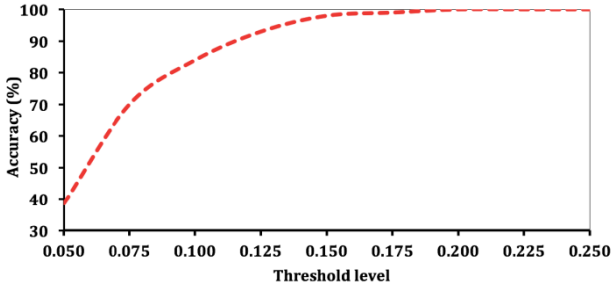

Fig. 7: Relationship between threshold and accuracy

5. Conclusion

Facial expression in images has been one of the interesting research topics in the field of image processing and widely employed in many practical applications. There are two major problems to be considered: (1) identifying facial expressions; and (2) demonstrating facial expressions, in which estimating facial expressions is the core issue. In this paper, we propose an estimation approach based on a random selection of geometric features via a decision tree. Through our empirical tests on a standard database, our approach provides satisfactory results which encourage us to have further research in imitating real human actions.

Acknowledgement

This research is supported by the Project VAST01.09/17-18 “Research and development of techniques to support Museum exhibits based on Virtual reality technology”.

References

Baker S and Matthews I (2001). Equivalence and efficiency of image alignment algorithms. In the Computer Vision and Pattern Recognition Conference, IEEE, Kauai, HI, USA: 1090-1097. https://doi.org/10.1109/CVPR.2001.990652

Bartlett M, Littlewort-Ford G, Frank M, Lainscsek C, Fasel I, and Movellan J (2006). Fully automatic facial action recognition in spontaneous behaviour. In the IEEE 7th International Conferance on Automatic Face and Gesture Recognition, Southampton, UK: 223–230. https://doi.org/10.1109/FGR.2006.55

Bashan S and Venayagamoorthy GK (2008). Recognition of facial expressions using Gabor wavelets and learning vector quantization. Engineering Applications of Artificial Intelligence, 21(7): 1056–1064.

Cootes TF, Edwards GJ, and Taylor CJ (2001). Active appearance models. IEEE Transactions on Pattern Analysis and Machine Intelligence, 23(6): 681–685.

Hien LT and Toan DN (2016). An algorithm to detect driver’s drowsiness based on nodding behaviour. International Journal of Soft Computing, Mathematics and Control, 5(1): 1-8.

Jiang B, Valstar M, and Pantic M (2011). Action unit detection using sparse appearance descriptors in space-time video volumes. In the IEEE International Conference Automatic Face and Gesture Recognition, IEEE, Santa Barbara, CA, USA: 314-321. https://doi.org/10.1109/FG.2011.5771416

Lucey S, Matthews I, Hu C, Ambadar Z, Torre FDL, and Cohn J (2006). AAM derived face representations for robust facial action recognition. In the 7th International Conference on Automatic Face and Gesture Recognition, IEEE, Southampton, UK: 155–160. https://doi.org/10.1109/FGR.2006.17

Oliveira LES, Koerich AL, Mansano M, and Britto ASJ (2011). 2D Principal component analysis for face and facial-expression recognition. Computing in Science and Engineering, 13(3): 9–13.

Tian Y, Kanade T, and Cohn J (2001). Recognizing action units for facial expression analysis. IEEE Trans. Pattern Analysis and Machine Intelligence, 23(2): 97-115.

Valstar MF & Pantic M (2007). Combined support vector machines and hidden markov models for modeling facial action temporal dynamics. In the International Workshop on Human-Computer Interaction, Springer, Berlin, Heidelberg: 118-127. https://doi.org/10.1007/978-3-540-75773-3_13

Viola P and Jones M (2001). Rapid object detection using a boosted cascade of simple features. In the IEEE Conference Computer Vision and Pattern Recognition, IEEE, Kauai, HI, USA: 511-518. https://doi.org/10.1109/CVPR.2001.990517

Xiao J, Baker S, Matthews I, and Kanade T (2004). Real-Time combined 2D+3D active appearance models. In the IEEE Conference on Computer Vision and Pattern Recognition, IEEE, Washington, DC, USA: 535-542. https://doi.org/10.1109/CVPR.2004.1315210

Zhao G and Pietikainen M (2007). Dynamic texture recognition using local binary patterns with an application to facial expressions. IEEE Transactions on Pattern Analysis and Machine Intelligence, 29(6): 915-928.