DOI: 10.2507/28th.daaam.proceedings.164

EMOTION RECOGNITION IN VIDEO WITH OPENCV AND COGNITIVE SERVICES API: A COMPARISON

Luis Antonio Beltrán Prieto & Zuzana Komínková Oplatková

This Publication has to be referred as: Beltran Prieto, L[uis] A[ntonio] & Kominkova Oplatkova, Z[uzana] (2017). Emotion Recognition in Video with OpenCV and Cognitive Services API: A Comparison, Proceedings of the 28th DAAAM International Symposium, pp.1185-1190, B. Katalinic (Ed.), Published by DAAAM International, ISBN 978-3-902734-11-2, ISSN 1726-9679, Vienna, Austria
DOI: 10.2507/28th.daaam.proceedings.164

Abstract

Emotions are people's reactions to certain stimuli. Most common way to detect an emotion is by facial expression analysis. Machine learning algorithms combined with other artificial intelligence techniques have been developed in order to identify expressions found in images and videos. Support Vector Machines, along with Haar Cascade classifiers can be used for efficient emotion recognition. OpenCV, an open-source library for machine learning, makes it possible to develop computer-vision applications. Cognitive Services is a free set of APIs which easily integrate artificial intelligence in applications. In this paper a comparison between two implementations of Emotion Recognition algorithms, namely SVM and Cognitive Services API, was carried out to compare their performance. For this research, 500 tests were performed per experiment. The SVM implementation in OpenCV obtained the best performance, with an 84% accuracy, which can be boosted by increasing the sample size per emotion.

Keywords: Support Vector Machine; OpenCV; Cognitive Services; Face Detection; Emotion Recognition; Haar Cascades

1. Introduction

Facial emotion detection can be defined as the process of recognizing the feeling that a person is expressing at a particular moment. Potential applications of emotion recognition include the improvement of student engagement [1], the built of smart health environments [2], the analysis of customers' feedback [3], and the evaluation of quality in children's games [4], among others. Face recognition within multimedia elements, such as images and videos, has been one of the challenges in the artificial intelligence field. Several powerful techniques have been examined exhaustively in the search of performance and accuracy improvement, for instance, Convolutional Neural Networks (CNN) [5], Deep Belief Networks (DBN) [6], and Support Vector Machines (SVM) [7], just to name a few. CNN and DBN are deep learning techniques. Deep Learning is a novel area of research in machine learning which focuses on learning high-level representations and abstractions of data, such as images, sound, and text by using hierarchical architectures, including neural networks, convolution networks, belief networks, and recurrent neural networks in several artificial intelligence areas, some of which are image classification [8], speech recognition [9], handwriting recognition [10], computer vision [11], and natural language processing [12].

Identifying the sentiment expressed by a person is one of the outcomes after achieving face detection. Recent research [13] has proven that emotion recognition can be accomplished by implementing machine learning and artificial intelligence algorithms. To assist in this task, several open-source libraries and packages, being OpenCV, TensorFlow, Theano, Caffe and the Microsoft Cognitive Toolkit (CNTK) some of the most notorious examples, cut down the process of building deep-learning-based algorithms and applications. Emotions such as anger, disgust, happiness, surprise, and neutrality can be detected.

The aim of this paper is to compare the performance of two emotion-recognition implementations from video sources. The first one is a Python-based application which uses OpenCV libraries with Support Vector Machine. The second implementation is a C# application which sends requests to the Emotion recognition API from Cognitive Services. 8000 facial expressions from the Radboud Faces Database were examined in different phases of the experiments for training and evaluation purposes.

This paper is organized as follows. Background information introducing emotion recognition, Support Vector Machines, OpenCV, Cognitive Services, and the Radboud Faces Database is presented firstly. Afterwards, the problem solution is described by explaining the methods and methodology that were used for this comparison. Evaluation results are shown subsequently. Finally, conclusions are discussed at the final section of the paper.

2. Background information

2.1. Emotion recognition

Emotions are strong feelings about people's situations and relationships with others. Most of the time, humans show how they feel by using facial expressions. Speech, gestures, and actions are also used to describe a person's current state. Emotion recognition can be defined as the process of detecting the feeling expressed by humans from their facial expressions, such as anger, happiness, sadness, deceitfulness, and others. Even though a person can automatically identify facial emotions, machine learning algorithms have been developed for this purpose. Emotions play a key role in decision-making and human-behaviour, as many actions are determined by how a person feels at some point.

Typically, these algorithms use either a picture or a video (which can be considered as a set of images) as input, then they proceed to detect and focus their attention on a face and finally, specific points and regions of the face are analysed in order to detect the affective state. Machine Learning algorithms, methods and techniques can be applied to detect emotions from a picture or video. For instance, a deep learning neural network can perform effective human activity recognition with the aid of smartphone sensors [14]. Moreover, a classification of facial expressions based on Support Vector Machines was developed for spontaneous behaviour analysis [15].

2.2. Support Vector Machines

Support Vector Machines were introduced [16] as a technique aimed to solve binary classification problems; due to their solid theoretical fundaments, SVMs have been used to answer regression, clustering and multi-classification tasks [17] along with practical applications in several fields, including computer vision [18], text classification [19], and natural language processing [20], among others. It can be defined as a discriminative classifier which works with labelled training data to output an optimal hyperplane used to categorize new examples.

While several learning techniques focus on minimizing the error rate generated by the model based on the training samples, SVMs attempt to minimize the so-called structural risk. The idea is to choose a separation hyperplane which is equidistant from the nearest examples of each class in order to obtain a maximum margin from each side of the hyperplane. Furthermore, when defining the hyperplane, only those training samples which are near the border of these margins are considered. These examples are known as support vectors. From a practical point of view, the maximum margin separating hyperplane has demonstrated to achieve a good generalization capacity, thus avoiding the overfitting of the training set.

Given a set of separable samples $S = \{(x_1, y_1), \ldots, (x_n, y_n)\}$, where $x_i \in \mathbb{R}^d$ and $y_i \in \{+1, -1\}$, a separation hyperplane, as shown in Fig. 1, can be defined as a linear function capable of split both sets without errors, according to (1), where w and b are real coefficients.

$$D(x) = (w_1 x_1 + \cdots + w_d x_d) + b = < w, x > +b \tag{1}$$

The separation hyperplane meets the constraints expressed in (2) for all x_i from the examples set.

$$\begin{cases} < w, x_i > +b \ \geq 0 & if \quad y_i = +1 \\ < w, x_i > +b \leq 0 & if \quad y_i = -1, i = 1, \ldots, n \end{cases} \tag{2}$$

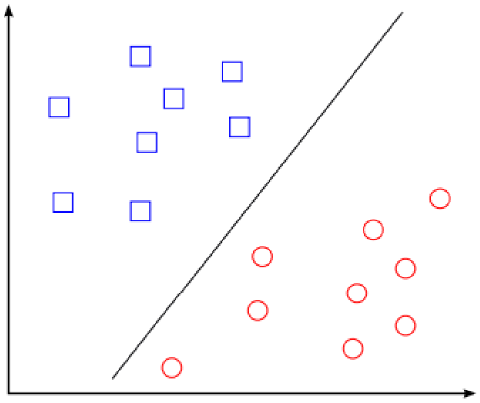


Fig. 1. Separation hyperplane in a bidimentional space from a set of examples separable in two classes

2.3. OpenCV

OpenCV [21] is a free, yet powerful, open-source library developed by Intel Corporation which has been widely used in computer vision and machine learning tasks, such as image processing, real-time image recognition, and face detection. With more than 2500 optimized algorithms included, this library has been extensively used for research and commercial applications from both global and small entrepreneurs. OpenCV contains an optimized set of libraries written in C language, with bindings to other languages and technologies, including Python, Android, iOS, and CUDA (for GPU fast processing), and wrappers in other languages, such as C#, Perl, Haskell, and others. Moreover, it works under Windows and Linux.

2.4. Cognitive Services

Cognitive Services [22] are a set of machine learning algorithms developed by Microsoft which are able to solve artificial intelligence problems in several fields, such as computer vision, speech recognition, natural language processing, machine learning search, and recommendation systems, among others. These algorithms can be consumed through Representational State Transfer (REST) calls over an Internet connection, allowing developers to use artificial intelligence research to solve problems. These services are open-source and can be consumed by many programming languages, including C#, PHP, Java, Python, and implemented in desktop, mobile, console, and web applications.

The Computer Vision API of Cognitive Services provides access to machine learning algorithms capable of performing image processing tasks. Either an image stream is uploaded or an image URL is specified to the service so the content can be analyzed for label tagging, image categorization, face detection, color extraction, text detection, and emotion recognition. Video is also supported as input. The Emotion API analyses the sentiment of a person in an image or video and returns the confidence level for eight emotions mostly understood by a facial expression, including anger, contempt, disgust, fear, happiness, neutrality, sadness and surprise.

2.5. The Radboud Faces Database

The Radboud Faces Database (RaFD) [23] is a high-quality set of pictures of 67 models (between male, female, Caucasian, Moroccan, children and adults) displaying 8 emotional expressions (anger, disgust, fear, happiness, sadness, surprise, contempt, and neutrality) in which each emotion was shown looking to three directions (left, front, and right), with five camera angles, as presented in Fig. 2. In total, the database contains 28709 faces. This initiative by the Behavioural Science Institute of the Radboud University Nijmegen is available for research purposes upon request.

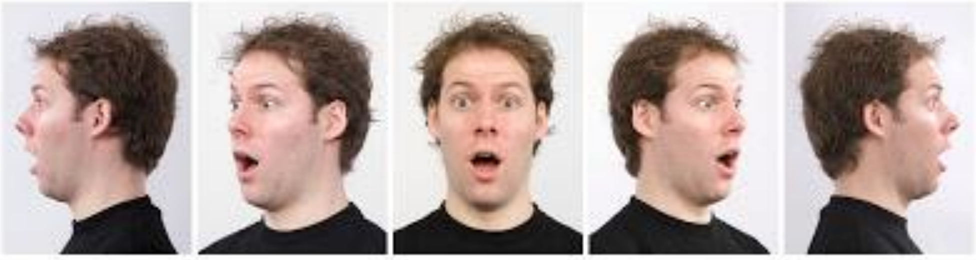


Fig. 2. Sample images from the RaFD

3. Methods and Methodology

The objective of this experiment is to compare the performance of a couple of emotion-recognition implementations in video. The first analysis (Experiment A) is a Python-based application which makes use of the OpenCV machine learning algorithms combined with Support Vector Machines for facial and emotion detection. The second study (Experiment B) is a C# mobile application which sends requests to a Cognitive Services API for emotion detection. In both cases, the Radboud Faces Database was used as input for the analysis.

For Experiment A, 8000 high resolution sequences which actually show a relevant expression, i.e. from a neutral feeling to the emotion itself, with 1000 images per emotion were taken into consideration. First step is then to obtain both the neutral and emotional faces. From this subset, OpenCV library is used to detect the face on each picture by using a custom Haar-filter. A successful object detection is possible because of the Haar feature-based cascade classifiers proposed in [24], which is an approach based on machine learning which involves a training from both positive and negative images, i.e. pictures with faces and without them, respectively. OpenCV already includes several Haar-filter libraries. Thereafter, all images were standardized by converting them to grayscale and resized to the same dimensions, an array of 48x48 grey-scale images. A Support Vector Machine was used in the training process of the classifier, consisting of the implementation of a non-linear support vector classification model with kernel radial basis function. The training process consists of getting the characteristics of each face along with the labelled emotion expressed by the person. Then, evaluation of the classifier proceeds by comparing the outcome of its predict function of each face with the actual labelled emotion. Fig. 2 presents the output of a video analysis of this experiment.

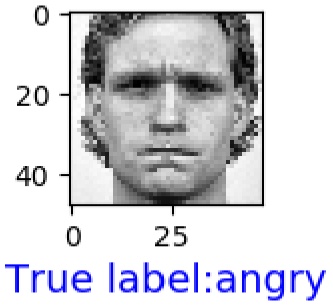
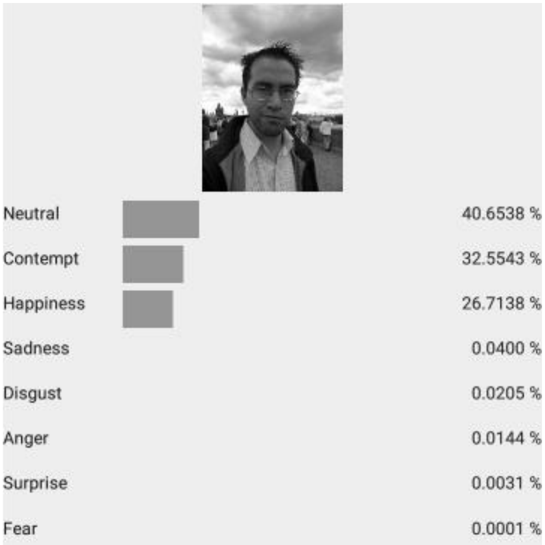


Fig. 3. Analysis of emotions detected on a video by the SVM implementation with OpenCV.

Experiment B starts with the 8000 images obtained from the dataset. 10 random groups consisting each of 500 sequences were generated for evaluation purposes. For each test, every video was submitted to the Cognitive Services API for its evaluation. A C# mobile application was developed with Xamarin cross-platform technology in order to execute this experiment. The service returns the information in a JSON-based format, containing the score for each emotion. The highest score means the facial expression detected by the service, and this result was compared with the actual labelled emotion for evaluation purposes. Fig. 4 shows the application developed in C# for this experiment. It extracts a picture from a video stream, then finds a face on it and finally sends a request to the service in order to detect the emotions expressed by the person. The analysis is performed by the Emotion Cognitive Service Video API.



Neutral	40.6538 %
Contempt	32.5543 %
Happiness	26.7138 %
Sadness	0.0400 %
Disgust	0.0205 %
Anger	0.0144 %
Surprise	0.0031 %
Fear	0.0001 %

Fig. 4. Analysis of emotions detected on a video by the Emotion Cognitive Service.

4. Results

After running each of the 10 tests from Experiment A, the results which are presented in Table 1 were obtained. An average of 84.02 % correct predictions was calculated as an outcome. Likewise, Table 2 illustrates the outcome of each test in Experiment B. As a result, a 68.93 % average efficiency was accomplished after running 10 tests of this implementation.

Test Number	Correct predictions (%)	Incorrect predictions (%)
1	405 (81.00 %)	95 (19.00 %)
2	430 (86.00 %)	70 (14.00 %)
3	407 (81.40 %)	93 (18.60 %)
4	416 (83.20 %)	84 (16.80 %)
5	429 (85.80 %)	71 (14.20 %)
6	428 (85.60 %)	72 (14.40 %)
7	410 (82.00 %)	90 (18.00 %)
8	426 (85.20 %)	74 (14.80 %)
9	418 (83.60 %)	82 (16.40 %)
10	432 (86.40 %)	68 (13.60 %)

Table 1. Evaluation results from Experiment A

Test Number	Correct predictions (%)	Incorrect predictions (%)
1	339 (67.93%)	161 (32.06%)
2	362 (72.51%)	138 (27.48%)
3	328 (65.64%)	172 (34.35%)
4	347 (69.46%)	153 (30.53%)
5	336 (67.17%)	164 (32.82%)
6	343 (68.70%)	157 (31.29%)
7	362 (72.51%)	138 (27.48%)
8	355 (70.99%)	145 (29.00%)
9	321 (64.12%)	179 (35.87%)
10	351 (70.22%)	149 (29.77%)

Table 2. Evaluation results from Experiment B

The findings of the experiments show that the Python-based implementation in OpenCV with Support Vector Machines returned a higher accuracy than the Cognitive Services implementation in C# by approximately a 15% difference. Minor mistakes occurred in Experiment A when trying to predict emotions that are similar, particularly fear and contempt, which were wrongly classified as sadness and neutral, respectively. Likewise, a neutral face most of the time was identified as a sad face.

This behaviour was also found in Experiment B, in which neutral emotions were spotted as either contempt or sadness; taking a look at the scores obtained by the Cognitive Services API, a minimal difference between the incorrect predicted emotion and the real one was detected. Thus, in most cases, the second-best prediction was correct. However, for evaluation purposes of this experiment, this incorrect assumption was considered as a failure.

5. Conclusions

The objective of this experiment was to compare the performance of a couple of different implementations of emotion-recognition applications in faces from videos by using OpenCV and Support Vector Machines in the first case, while considering a C#-based solution which sends requests to a Cognitive Services API for Emotion detection in video for the second solution.

While the first implementation got the best results, the performance could be improved by increasing the sample size of those emotions with few faces, so the training phase gets benefited. Future research will include emotion recognition in faces performed by Deep Learning techniques, including Convolutional Neural Networks and Deep Belief Networks to name a few, which while more complex, are also more suitable for difficult tasks in terms of both performance and computational time.

6. Acknowledgments

This work was supported by the Ministry of Education, Youth and Sports of the Czech Republic within the National Sustainability Programme project No. LO1303 (MSMT-7778/2014) and also by the European Regional Development Fund under the project CEBIA-Tech No. CZ.1.05/2.1.00/03.0089, further it was supported by Grant Agency of the Czech Republic—GACR 588 P103/15/06700S and by Internal Grant Agency of Tomas Bata University in Zlin under the project No. IGA/CebiaTech/2017/004. L.A.B.P author also thanks the doctoral scholarship provided by the National Council for Science and Technology (CONACYT) and the Council for Science and Technology of the State of Guanajuato (CONCYTEG) in Mexico.

7. References

[1] Garn AC, Simonton K, Dasingert T, Simonton A. Predicting changes in student engagement in university physical education: Application of control-value theory of achievement emotions, Psychology of Sport and Exercise, Vol.29, pp. 93-102.

[2] Fernandez-Caballero A, Martinez-Rodrigo A, Pastor JM, Castillo JC, Lozano-Monasor E, Lopez MT, Zangroniz R, Latorre JM, Fernandez-Sotos A. Smart environment architecture for emotion detection and regulation, Journal of Biomedical Informatics, Vol.64 pp-57-73.

[3] Felbermayr A, Nanopoulos A. The Role of Emotions for the Perceived Usefulness in Online Customer Reviews, Jounal of Interactive Marketing, Vol.36, pp. 60-76.

[4] Gennari R, Melonio A, Raccanello D, Brondino M, Dodero G, Pasini M, Torello S. Children's emotions and quality of products in participatory game design, International Journal of Human-Computer Studies, Vol.101, pp. 45-61.

[5] LeCun Y, Bengio Y. Convolutional networks for images, speech, and time-series. The Handbook of Brain Theory and Neural Networks. MIT Press, 1995.

[6] Hinton GE, Osindero S, The Y. A Fast Learning Algorithm for Deep Belief Nets. Neural Computation. Vol. 18, 2016 No. 7 pp. 1527-1554

[7] Chen D, Tian Y, Liu X. Structural nonparallel support vector machine for pattern recognition. In Pattern Recognition, Vol. 60, 2016, pp. 296-305

[8] Zhang Y, Zhang E, Chen W. Deep neural network for halftone image classification based on sparse auto-encoder, In Engineering Applications of Artificial Intelligence, Vol 50, 2016. pp. 245-255

[9] Huang Z, Siniscalchi SM, Lee C. A unified approach to transfer learning of deep neural networks with applications to speaker adaptation in automatic speech recognition, In Neurocomputing, Vol. 218, 2016. pp. 448-459

[10] Elleuch M, Maalej R, Kherallah M. A New Design Based-SVM of the CNN Classifier Architecture with Dropout for Offline Arabic Handwritten Recognition, In Procedia Computer Science, Vol. 80, 2016. pp. 1712-1723

[11] Liu H, Lu J, Feng J, Zhou J. Group-aware deep feature learning for facial age estimation, Pattern Recognition, Vol.66, pp. 82-94.

[12] Bayer AB, Riccardi G. Semantic language models with deep neural networks, In Computer Speech & Language, Vol. 40, 2016, pp. 1-22

[13] Yogesh CK, Hariharan M, Ngadiran R, Adom AH, Yaacob S, Polat K, Hybrid BBO_PSO and higher order spectral features for emotion and stress recognition from natural speech, Applied Soft Computing, Vol.56, pp. 217-232.

[14] Ronao CA, Cho S, Human activity recognition with smartphone sensors using deep learning neural networks, Expert Systems with Applications, Vol.59, pp. 235-244.

[15] Bartlett MS, Littlewort G, Frank M, Lainscsek C, Fasel I, Movellan J, Recognizing facial expression: machine learning and application to spontaneous behavior. 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05). Vol. 2. pp. 568-573

[16] Boser BE, Guyon IM, Vapnik VN. A training algorithm for optimal margin classifiers. In Proceedings of the 5th Annual Workshop on Computational Learning Theory. pp. 144-152

[17] Ding S, Zhang X, An Y, Xue Y. Weighted linear loss multiple birth support vector machine based on information granulation for multi-class classification, In Pattern Recognition, Vol. 67, 2017. pp. 32-46

[18] Cha Y, You K, Choi W. Vision-based detection of loosened bolts using the Hough transform and support vector machines. In Automation in Construction. Vol. 71, Part 2, 2016, pp. 181-188

[19] Ramesh B, Sathiaseelan JGR. An Advanced Multi Class Instance Selection based Support Vector Machine for Text Classification. In Procedia Computer Science. Vol. 57, 2015, pp. 1124-1130

[20] Yang H, Lee S. Robust sign language recognition by combining manual and non-manual features based on conditional random field and support vector machine. In Pattern Recognition Letters. Vol. 34, Issue 16, 2013. pp. 2051-2056.

[21] OpenCV library. http://opencv.org [Online: accessed 01-Jun-2017]

[22] Cognitive Services – Intelligence Applications. http://microsoft.com/cognitive [Online: accessed 03-Jun-2017]

[23] Langner O, Dotsch R, Bijlstra G, Wigboldus DHJ, Hawk ST, van Knippenberg A. (2010). Presentation and validation of the Radboud Faces Database. Cognition & Emotion. Vol. 24 No. 8. pp. 1377-1388.

[24] Turk M, Pentland A. Eigenfaces for Recognition. Journal of Cognitive Neuroscience. Vol.3, No. 1, pp. 71-86.