Human Computer Interaction

Polytechnic University of Catalonia

Barcelona

Report of the project

EmotiNet

Khizer Amin

Nariman Nakhjiri

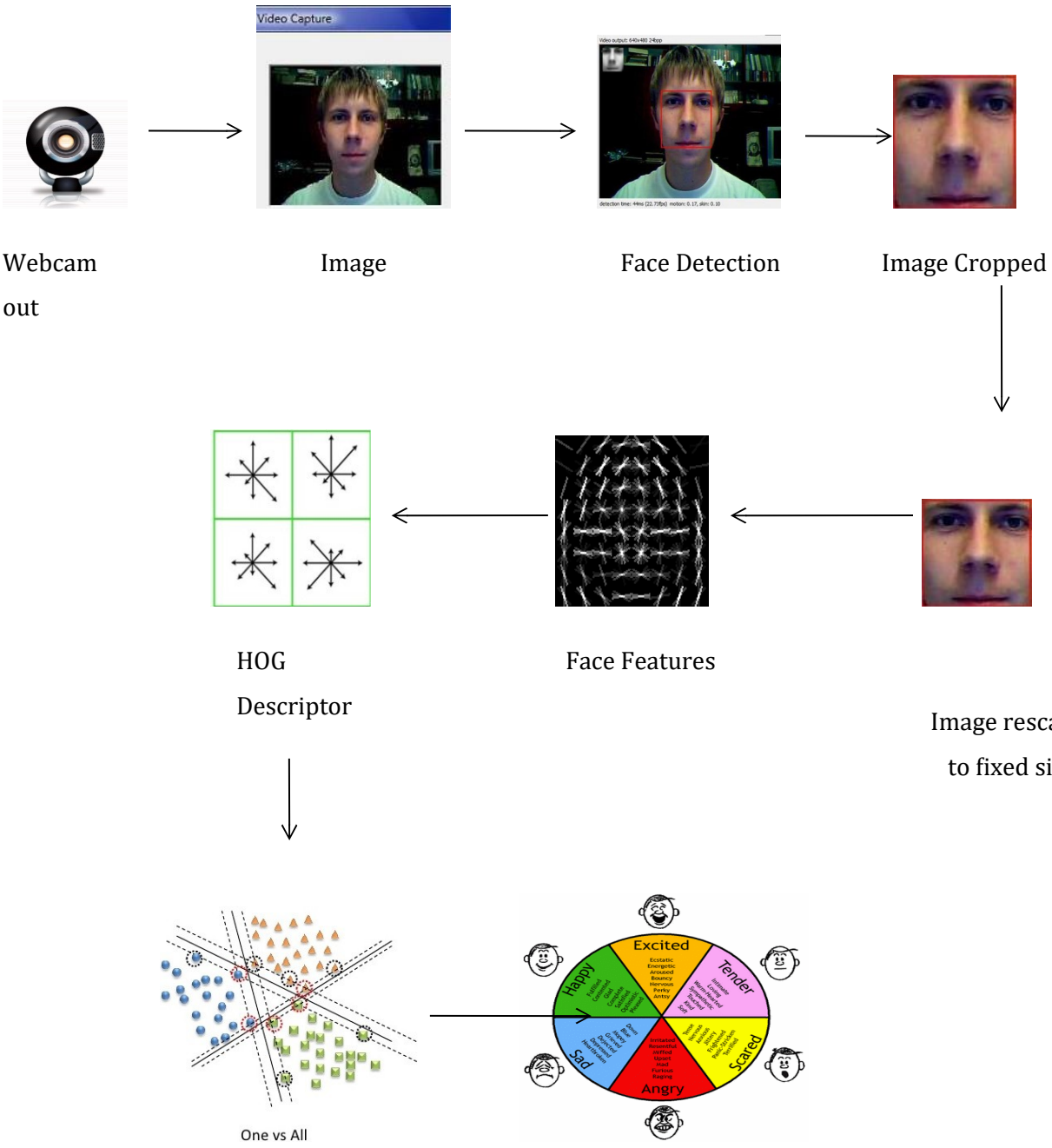
Nima Sanjabi

Feb-June 2016

Introduction

EmotiNet is a messaging application, aimed to bring a more natural and realistic communication experience to the users. The idea is to get the emotions of the user, like anger, happiness, sadness etc. and send them in a messaging application attached to the text messages. The application will use computer vision, and face recognition techniques to capture the user emotions and feelings. User emotions will automatically be converted to corresponding colors, or smileys, and will be attached to the text messages.

Emotions recognition pipeline and processing



Webcam out

Image

Face Detection

Image Cropped

Image rescaled to fixed size

Face Features

HOG Descriptor

SVM Classifier

Emotion

Figure A. Facial emotions recognition pipeline

For the application we trained 524 images, the model showed an accuracy of 99 *percent*. The link to the images and feature database is provided as a reference [1]. Also we used OpenCV ver-2.4.10 for our project, OpenCV is a standard Computer vision library.

The Figure A above exhibits all the aspects of an emotions recognition processing, the figure also represents the core of EmotiNet project, which is to recognize facial expressions and come up with a final label. The following steps briefly discuss all the steps involved in the pipeline.

a) Webcam

For the project, we used a simple computer mounted video camera to capture the frames from it at real time. We used OpenCV's VideoCapture library for this purpose.

b) Image

The retrieved image from the video camera should represent full face a person, and the all over pixel quality of the image should be better. Modern day PC camera's can provide video streams with good resolution. This requirement is to train the images which has good pixel resolution.

c) Face Detection

The Face will be detected next from the retrieved image. For this purpose, we used famous HAAR cascade object detection classifier proposed by Viola-Jones in 2001. HAAR calculates features of the face as rectangular regions, the sums of pixel values over rectangular regions are calculated rapidly using *integral images.*

d) Crop the face

Next step in the pipeline is to crop the face out of the detected image. Again OpenCV helped with its own built-in function for selecting a section of an image. Cropping the face is an important part for computation purposes, this way we can detect features from the face only and our software does not have to look for non-face section of the whole image.

e) Rescale the image

In this step the cropped out face image is rescaled to a fixed length size. Feature vector that we detected from image should be of constant size for both training/testing.

f) Face Features

From those rescaled images, the significant features will be selected. The training images used for the experiment has more 3000 features and dimensions, but for to make systems run faster and smoother we have to reduce the feature length. For dimensionality reduction of those face features we used PCA (Principal component analysis), a well-known dimensionality reduction feature used in most of the modern day Machine Learning applications. OpenCV has provided a building PCA function for that purpose, Reducing the feature length has actually saved our computation time for training a huge number of features.

g) HOG Descriptor

HOG is histogram of oriented gradients, is a feature descriptor used in computer vision for object detection in the images. It is a descriptor of the all the features that is detected in the previous stage, it contains features vectors with their orientations. The descriptor will further used in the emotion classifier.

h) SVM classifier

SVM, the support vector machine is a supervised learning technique used for classification, the advantage of using SVMs is that they can be effective in high dimensions. SVMs are used along with different kernel functions, for our application we used SVM with a linear kernel, i.e. **(x, x').** We performed a multi-class classification.

The decision function is:

$$\operatorname{sgn}(\sum_{i=1}^{n} y_i \alpha_i K(x_i, x) + \rho)$$

i) Emotions

Finally, the classifier comes up with the final label, which is an emotion. For the purpose of our experiment and also to make the allover training time smaller, we used these classes of emotions represented as an integer value, i.e.: Happy, Anger, Surprise and a label for a Neutral face.

Technologies Used

C++ and OpenCV 2.4.10 Library

Java.net

For the server and communication. And also for development of the GUI.

Challenges

One of the main challenges might be that, the color expressions would not satisfy the users. This problem can be solved using user feedbacks;
It is possible to gather information about the user experiences with the interface, and to find out what colors do better convey a specific emotion. Of course the higher the user feedbacks, we can better decide on the improvements.

We also need to set the emotion thresholds accurately, and to cope with noise, and rapid face movements.

computation problem; the real time Image capturing and face detection, demands a high level of computation power. We have solved it by scaling down the camera window

emotion recognition is trained using a huge dataset of real face gestures.

User Feedback

we have presented our application two 7 individuals and asked them how they think about its functionality. The results are shown in table 1.

	General Idea	Learnability	Ease of Access	Usefulness	Enjoyable
Mozzezi.M	8	7	8	7	7
Ekechi.L	8	6	9	6	7
Farzi.A	9	-	9	10	9
Antonio.J	7	8	-	9	8
Azad.R	6	6	7	5	4
Andre.M	9	-	8	8	8
Sara.N	8	7	9	7	7

Table 1. User Feedbacks

Future Work

The prospects for this EmotiNet are huge, and there are number of direction we can go towards. One of the other things is to export the whole platform to mobile devices, such as android devices and iPhones. We think if we can further reduce the computation time for training images, the device can easily work on any mobile surface, just think if our cell phones can capture and understand our emotions, and then communicate them over networks, and how this idea can bring about changes in people's behavior towards each other.

Not just mobiles, we can transport the technology to other devices as well, for example we can implement emotion recognition in devices that we daily use, i.e. devices can run merely according to our current mood and emotion. The first thing that comes in mind is *IOT,* i.e. internet of things, which change our day to day lives. Implementing the emotion recognition methods in IOE systems can be a really exciting future work.

Research is already going on in emotion recognition techniques and computer scientist are finding ways to take advantage from this innovative idea of recognizing human's current state of emotions.

References

[1]. http://www.consortium.ri.cmu.edu/ckagree/

[2]. opencv.org/downloads.html