Neural Networks 72 (2015) 140–151

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

2015 Special Issue

Multimodal emotional state recognition using sequence-dependent deep hierarchical features

Pablo Barros [*], Doreen Jirak, Cornelius Weber, Stefan Wermter

Department of Informatics, University of Hamburg, Knowledge Technology, Vogt-Koelln-Strasse 30, 22527 Hamburg, Germany

ARTICLE INFO

Article history:
Available online 19 October 2015

Keywords:
Emotion recognition
Deep learning
Convolutional Neural Networks
Hierarchical features
Human Robot Interaction

ABSTRACT

Emotional state recognition has become an important topic for human–robot interaction in the past years. By determining emotion expressions, robots can identify important variables of human behavior and use these to communicate in a more human-like fashion and thereby extend the interaction possibilities. Human emotions are multimodal and spontaneous, which makes them hard to be recognized by robots. Each modality has its own restrictions and constraints which, together with the non-structured behavior of spontaneous expressions, create several difficulties for the approaches present in the literature, which are based on several explicit feature extraction techniques and manual modality fusion. Our model uses a hierarchical feature representation to deal with spontaneous emotions, and learns how to integrate multiple modalities for non-verbal emotion recognition, making it suitable to be used in an HRI scenario. Our experiments show that a significant improvement of recognition accuracy is achieved when we use hierarchical features and multimodal information, and our model improves the accuracy of state-of-the-art approaches from 82.5% reported in the literature to 91.3% for a benchmark dataset on spontaneous emotion expressions.

1. Introduction

Human emotional expression recognition has been the focus of attention in several areas from psychology and neuroscience to cognitive and computer science due to its importance in human communication, interaction and social relations. On the one hand, Aggarwal and Ryoo (2011) show that human actions can be well recognized only based on motion and appearance, ignoring the emotion expressions. On the other hand, there is important additional information in the emotional expressions of humans, for example when the context of a dialogue is changed based on irony or sarcasm. When applied to human–machine interaction the understanding of emotions could be used to improve dialogues, predict human behavior and plan a human-like reaction, as shown in the research of Morishima and Harashima (1993). They presented a pioneering use of emotions in a human–machine interaction scenario, where a virtual avatar changes its facial expression based on the emotional tones present in the human voice. Emotion expression recognition can also be applied to expert systems; Tokuno et al. (2011) use emotion determination in a medical scenario. In their system, a robot is used to help in the diagnosis of mental stress based on the patient's speech tone.

The consideration and application of emotions for Human Robot Interaction (HRI) are important for the acceptance of robots by humans. A robot that is able to recognize and express emotions can communicate in a natural manner. Such a robot can learn different emotions from humans, and identify when it is appropriate to use them, as shown in the work of Gadanho (2003). Lerner and Keltner (2000) showed that emotions are important factors in decision and judgment tasks. A robot that recognizes emotions can be deployed as a human teammate in different social tasks, as discussed by Breazeal and Brooks (2004). Spexard, Hanheide, and Sagerer (2007) apply emotion recognition to improve the dialogues of an anthropomorphic robot in a HRI scenario, showing that when emotion is taken into consideration, the interaction with the robot is more natural.

As discussed by Cabanac (2002), there is no consensus in the literature to define emotional states, but the observation of several characteristics, and among them facial expressions, are used to identify them. Facial expressions are included in most of the emotional demonstrations, and are the characteristics which present the strongest impact when inferring the emotional state of a human. Based on the importance of facial expressions for emotion determination, Ekman and Friesen (1971) described

[*] Corresponding author.
E-mail addresses: barros@informatik.uni-hamburg.de (P. Barros), jirak@informatik.uni-hamburg.de (D. Jirak), weber@informatik.uni-hamburg.de (C. Weber), wermter@informatik.uni-hamburg.de (S. Wermter).

http://dx.doi.org/10.1016/j.neunet.2015.09.009

face expressions as universal, culturally and racially invariant emotion characteristics. In their work, they establish six universal emotions: "Disgust", "Fear", "Happiness", "Surprise", "Sadness" and "Anger". Later, Ekman and Friesen (1978) developed a system to measure facial expressions, denominated Facial Action Coding System (FACS). This system is used as a basis for several works in computer science in the past decades, as for example the work of Velusamy, Kannan, Anand, Sharma, and Navathe (2011). In their work they extract Action Units (AUs), described in the FACS as measurement units extracted from some face areas, and map them to six different emotions. Their method uses a series of templates to match the AUs with one of the six emotions. Although Velusamy et al. (2011) show good results on different datasets, this method is extremely dependent on the position of the face, illumination and distance of the camera to achieve good results.

Although face expression is essential in emotion determination, other characteristics should not be ignored, as shown in the work of Kret, Roelofs, Stekelenburg, and de Gelder (2013). They perform a psychological analysis on the observation of the whole body for emotional recognition. They show that face expression alone may contain misleading information, especially when applied to interaction and social scenarios. The observation of different modalities, such as body posture, motion, and speech intonation, improved the determination of the emotional state of the subjects. In congruence with this social experiment, a computer science approach for an automatic emotion recognition system based on multimodal information was proposed by Castellano, Kessous, and Caridakis (2008). They process facial expression, body posture and speech, extracting a series of features from each modality and combining them into one big feature vector. Although they show that when the different modalities are processed together, they present a better recognition accuracy, the extraction of each modality individually does not model the correlation between them, which could be found when processing the modalities together as one stream. Extracting features from all modalities at once would create a different feature representation that would not be constrained with the limitations of each individual modality.

The psychological study of Gu, Mai, and Luo (2013) analyzes the importance of the information in each modality present in human emotional states. They discuss that in non-verbal communication, facial expressions and body posture/motion complement each other when determining emotional states. In this respect, the observation of both modalities could provide a better accuracy in emotion definition. They show that when presented together, both modalities are recognized differently when they are shown individually, which shows a dependency within the modalities that changes the emotional recognition.

Gunes and Piccardi (2009) evaluate the efficiency of face expression, body motion and a fused representation for an automatic emotion recognition system. They realize two experiments, each one extracting specific features from face and body motion from the same corpus and compare the recognition accuracy. For face expressions, they track the face, and extract a series of features based on face landmarks. For body motion, they track the position of the shoulders, arms and head of the subject and extract 3522 feature vectors, using dozens of different specific feature extraction techniques. These feature vectors are classified using general classification techniques, such as Support Vector Machines and Random Forests. At the end, they fuse all feature vectors extracted from both experiments and classify them. The results obtained when fusing face and motion features were better than when these modalities were classified alone. The same conclusion was achieved by Chen, Tian, Liu, and Metaxas (2013). They apply a series of techniques to pre-process and extract specific features from face and body motion, similarly to Gunes and Piccardi (2009). Differences are that they use fewer features in the final representation and the time variance representation is different in both approaches. Gunes and Piccardi (2009) use a frame-based-classification, where each frame is classified individually and a stream of frames is later on scored to identify which emotional state is present. Chen et al. (2013) analyze two temporal representations: one based on a bag-of-words model and another based on a temporal normalization based on linear interpolation of the frames. Both works use the same solution based on manual feature fusion, which does not take into consideration the inner correlation between face expression and the body motion, but fused both of these features using a methodical scheme.

The work of Adolphs (2002) illustrates how the human brain recognizes emotions from visual stimuli, correlating information from different areas. The brain correlates past experiences, motion information in the visual stimuli, and face expressions. The brain is also capable to integrate this multimodal information and generate a representation for the visual stimuli based on all of them together. The simulation of this process in computer systems can be achieved by neural models, particularly ones which are able to create a hierarchy of feature representations such as Convolutional Neural Networks.

Convolutional Neural Networks (CNN) were introduced formally by Lecun, Bottou, Bengio, and Haffner (1998). They are inspired by the hierarchical process of simple and complex cells in the human brain to extract and learn different information from visual stimuli. Each layer of a CNN has the capability to react to different information, and when stacked together the layers can create a complex representation of the visual input. The first layers act as edge-like detectors, represent the simplest information of the visual stimuli: the differentiation of borders and contrasts. Passing this information to deeper layers, more complex representations are found. At deeper layers, representations of shape, orientation, position and image transformation are encoded. Due to the large variety of visual representation that these models could achieve, they were applied in several tasks. Fasel (2002) applies a CNN for face expression recognition. They present patches of different sizes to the network, and the network creates a representation of each patch individually and integrates them in the last layer. They were able to create a general feature representation for facial expressions simulating the FACS system. However, their model can only extract information from separated parts of the face at a time and sum them at the end to represent the facial expression. This process does not take into consideration the interactions and relations between each observed characteristic.

Based on the multimodal representation of visual stimuli described by Gu et al. (2013), and in the hierarchical representation present in the human brain as discussed by Adolphs (2002), we propose a novel CNN-based model for automatic emotion recognition. Our model extends the hierarchical visual representation of the CNN, and applies it for multimodal emotional state recognition using face expression and body motion. The proposed system is capable to generate a specific edge-like representation for each stimulus in the first layers and a complex representation of emotional states in the deeper ones. The use of hierarchical features allows the model to deal with spontaneous emotions. Each layer is capable to extract relevant information from a non-structured emotion expression, being able to identify which characteristics of the input stimuli is the most important. The first layers learn how to extract macro regions which contain important information, for example ignoring the background. Deeper layers learn how to extract detailed structures like eyes, mouth, hand movements and so on. This way, we do not rely on several different feature extraction techniques, but let the model learn in each layer which are the most relevant features. In the same way, each layer can also extract information from different characteristics, for example movement of the hands and the head, or position of the eye or mouth. By learning which characteristic is more relevant, our model passes them to deeper layers, and thus realizes a learned modality fusion.

To evaluate the correlation between the different modalities, three experiments were designed. The first one uses the proposed model to learn characteristics from facial expressions only, the second one from motion only and the third one from face expression and motion together. Different from previous approaches, our proposed system can create a shared representation of both modalities, without relying on manual feature fusion techniques. We understand manual fusion techniques as the ones that do not have an adaptation mechanism, but a hard-coded feature fusion strategy. The use of these techniques for classification tasks does not provide a big improvement when compared to classifying representations based on single modalities, as discussed by Wagner, Andre, Lingenfelser, and Kim (2011). Our shared representation is learned by the network, and consists of a new representation of both modalities. We do not have the sum of constraints of the individual modalities, as in the case of manual feature fusion approaches. The works of Chen et al. (2013) and Gunes and Piccardi (2009) use several different feature extraction techniques, and perform the feature fusion by uniting all features in one feature vector. The restrictions of each technique are accumulated, reducing the capability of generalization of their models. Our model learns how to fuse each multimodal stream by using a fully connected layer, removing the constraints of each stream and generating a set of robust features. Our results show that our approach is more appropriate for spontaneous emotion recognition than the models using manual feature fusion.

We evaluate the proposed model with the Gunes and Piccardi (2006) corpus. This corpus contains recordings of different subjects showing 10 different spontaneous emotions. The corpus is composed of different videos, comprising one to three sequences of subjects demonstrating emotional states in the same video. The proposed model is compared with literature solutions based on manual feature fusion techniques and the difference of feature representation is analyzed and discussed.

This paper is organized as follows: The next section shows the proposed model, describing how it deals with the three types of different stimuli. Section 3 describes the three experiments, parameter exploration and how the behavior of the network is analyzed. Section 3 also shows the results of our experiments and compares them with state-of-the-art solutions. A discussion of the results, the role of the parameters in the proposed model and on the importance of having a correlation between the modalities is given in Section 4. The conclusions and future work are presented in Section 5.

2. Proposed system

Our proposed model receives as input a continuous video stream. The frames of the video stream are used as input, and for each sequence a label for an emotional state is produced. Our model is implemented as a Multichannel Convolutional Neural Network (MCCNN) to extract hierarchical features from visual stimuli. After training, the model is able to extract edge-like features in the first layers and to generate a complex representation in deeper ones. These complex representations vary depending on the visual stimuli, showing different activation when a face or the whole body is presented.

To be able to recognize three different stimuli, the input stimuli are adapted for each realized experiment. The network's hierarchical topology is kept the same during all three experiments, but the input stimuli change. This shows the capacity of the model to learn from different inputs, and provides a robust evaluation scheme since the only aspect changing is the presented stimuli.

To be able to deal with sequences, a cubic receptive field implementation is used. This implementation is based on the work of Ji, Xu, Yang, and Yu (2013) and expands the CNN capability into modeling dependencies between frames. Our proposed model is able to learn simple and complex features and to model the dependencies of these features in a sequence. By using the multichannel implementation, it is also possible to define different features for the lowest level of the feature learning, specifically gray scale, Sobel X and Sobel Y filters. This capability improves the capacity of the network to adapt to each stimulus, and learn hierarchical features that are unique for each modality. That generates an exclusive and independent feature representation for each modality.

2.1. Convolutional neural network

In a CNN each layer is composed of two operations: convolution and max-pooling. These two operations simulate the response of simple and complex cell layers discovered in visual area V1 by Hubel and Wiesel (1959). The Neocognitron, proposed by Fukushima (1980), was the first deep neurocomputational model based on the simple and complex cells, inspiring the CNNs. In a CNN, the simple cell abstraction is represented by the convolution operations that use local filters to compute high-order features from input images. The complex cell abstraction increases the invariance by pooling simple cell units from the same receptive field from previous layers.

To increase the capability of the simple cells for extracting features, in each layer a series of different filters is applied to the image. This operation generates different images or filter maps, one for each filter, which are passed through all layers of the network. The complex cells act in each of these images generating spatial invariance for each filter.

Each set of filters in the simple cell layers acts in a receptive field in the image. The activation of each unit v_{nc}^{xy} at (x, y) of the nth filter in the cth layer is given by

$$v_{nc}^{xy} = \max\left(b_{nc} + \sum_{m}\sum_{h=1}^{H}\sum_{w=1}^{W} w_{(c-1)m}^{hw} v_{(c-1)m}^{(x+h)(y+w)}, 0\right), \tag{1}$$

where $\max(\cdot, 0)$ represents the rectified linear function, which was shown to be more suitable for training deep neural architectures, as discussed by Glorot, Bordes, and Bengio (2011), b_{nc} is the bias for the nth feature map of the cth layer, m indexes over the set of feature maps in the $(c-1)$ layer connected to the current layer c. $w_{(c-1)m}^{hw}$ is the weight of the connection between the unit (h, w) within a receptive field, connected to the previous layer, $c-1$, and to the filter map m. H and W are the height and width of the receptive field.

In the complex layers, a receptive field of the previous simple cell layer is connected to a complex cell in the current layer, reducing the dimension of the feature maps. The complex cell outputs the maximum activation of the receptive field $u(x, y)$ and is defined as

$$a_j = \max_{n\times n}\left(v_{nc} u(x, y)\right), \tag{2}$$

where v_{nc} is the output of the simple cell. In this function, the complex cell computes the maximum activation among the receptive field $u(x, y)$. The maximum operation down-samples the feature map, maintaining the simple cell structure.

The parameters of a CNN could be learned either by a supervised approach tuning the filters in a training database, as presented by Hinton, Osindero, and Teh (2006), or an unsupervised approach as present in the approach of Ranzato, Huang, Boureau, and LeCun (2007). Our proposed model uses the supervised approach. Although many approaches use unsupervised training for learning hierarchical features, like the research of Le, Zou, Yeung, and Ng (2011) and Ramirez-Amaro et al. (2013), most of the approaches for temporal hierarchical features learning using CNNs use supervised training, as for example the research of Karpathy et al. (2014) and Wang, Liu, Wang, Chan, and Yang (2015). Even for approaches where unsupervised training is used, such as deep belief networks or stacked auto encoders, the use of supervised fine tuning is advised, as discussed by Erhan et al. (2010). The use of supervised training allows us to train the network with a smaller amount of data, which would not be possible when using unsupervised training.

2.2. Sequence processing with cubic receptive fields

In a CNN the simple cells are applied on feature maps to compute spatial features. To create a sequence dependency between these features the concept of a cubic receptive field is applied to a stack of images. This concept was applied in the research of Ji et al. (2013) for action recognition. The value of each unit (x, y, z) at the nth filter map in the cth layer is defined as

$$v_{nc}^{xyz} = \max\left(b_{nc} + \sum_{m}\sum_{h=1}^{H}\sum_{w=1}^{W}\sum_{r=1}^{R} w_{(c-1)m}^{hwr}\, v_{(m-1)}^{(x+h)(y+w)(z+r)},\, 0\right) \quad (3)$$

where $\max(\cdot, 0)$ represents the rectified linear function, b_{cn} is the bias for the nth filter map of the cth layer, m indexes over the set of feature maps in the $(c-1)$ layer connected to the current layer c. In Eq. (2), $w_{(c-1)m}^{hwr}$ is the weight of the connection between the unit (h, w, r) within a receptive field connected to the previous layer $(c-1)$ and the filter map m. H and W are the height and width of the receptive field and z indexes each image in the image stack, R is the number of images stacked together representing the new dimension of the receptive field.

The same unit is connected to a sequence of images but always to the same region in each image. This gives the model the capacity to highlight pixel intensity variation for each class representation in the images, and allows the unit to learn the similar pixel intensities present in the stack of images. The tuning is improved by the fact that the connection is not shared between images, but each region in each image has its own weighted connection with the unit. This operation enhances the invariant responses within the same class, presenting the model with different pixel intensities in the same region of different images. The weights are tuned to learn this invariant response. A similar process can be found in the research of Wallis, Rolls, and Földiák (1993) and Wiskott and Sejnowski (2002). The cubic receptive field is applied only in the first convolutional layer that is connected directly with the input images.

2.3. Multichannel implementation

One of the problems with deep neural networks is the large amount of computational resources and time for training. To learn filters which are capable to extract meaningful information from an image, several layers and training epochs are necessary. The CNN implementation of Ji et al. (2013) uses fixed kernels to increase the complexity of the extracted features, reducing the number of parameters to be trained and the time necessary to train them. They embed the network with a series of filter maps with fixed weights, each one of them enhancing different properties of the image. Our model extends this process by implementing a multichannel architecture. We implement three channels, each one containing a CNN. The last layer of each CNN is fully connected to hidden layer, which produces the input for a logistic regression classifier. Each of our three channels represents specialized information and our model does not need to be so deep, reducing the number of parameters to be updated. The specialized information comes from the application of Sobel-based filters in the first layers of the channels, which do not have to be trained, and extract edges in two orientations: vertical and horizontal.

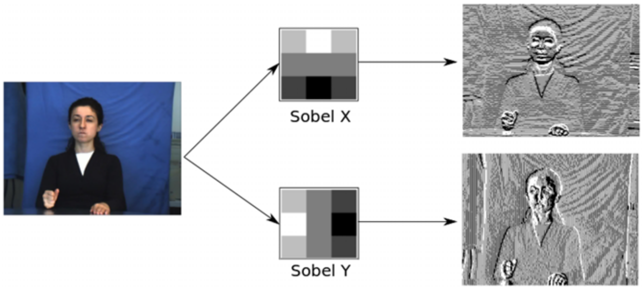


Fig. 1. Application of the Sobel filters in one image. It is possible to see how the filter enhances the image in vertical and horizontal directions.

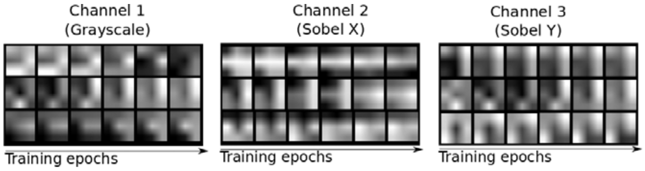


Fig. 2. Filters training through time. For each channel, different structures tend to emerge during training. Channels 2 and 3 show structures similar to Sobel filters, and channel 1 exhibits different and more complex structures.

Each of the channels of our network receives different information. In the first layer, two of the channels implement the previously mentioned untrained Sobel-based filters, each one in a different direction. The Sobel filters are very simple edge enhancement processes, and act here as an encoded edge enhancers. In most CNNs, after training, the first layers will become edge-detectors, in most of the cases similar to Gabor filters. When applying the Sobel operators in two of the channels, and training the network, each channel will be influenced by the other. The Sobel filters enhance very specific contrasts of the image and this affects the third channel. Each Sobel filter enhances the contrast between pixels on horizontal and vertical directions. Fig. 1 shows an example of the application of Sobel filters in one image. After training, the third channel will have developed a more complex edge representation than Gabor filters. This representation turns out to be very specific for the kind of stimulus that is presented to the network, which improves the final representation obtained by the model even when using fewer layers.

The three channels influence each other, driving the filters' training to a different direction when only one channel is trained. They share the same training process, and although the weight updates in each channel are individual, the fact that they are connected in the end creates a bias for the update. Fig. 2 illustrates some of the filters of the first layer of the network and their evolutions through each training epoch. We can see that the channels 2 and 3 produced edge-detectors with a similar structure of the Sobel filters, and this structure emerged during training. Channel 1 produced more complex filters, with structures similar to both channels combined.

As described in the work of Bar (2007), the human brain is continuously generating simple and rudimentary predictions that are used for a focused identification of visual stimuli. The implementation of the Sobel-filters in our architecture simulates this behavior, by using a simple edge-like enhancement to drive the learning of features by the model. The Sobel filters present this rudimentary information, and help the model creating a more complex and specific representation of the visual stimuli, accelerating the learning process. Fig. 3 illustrates the proposed

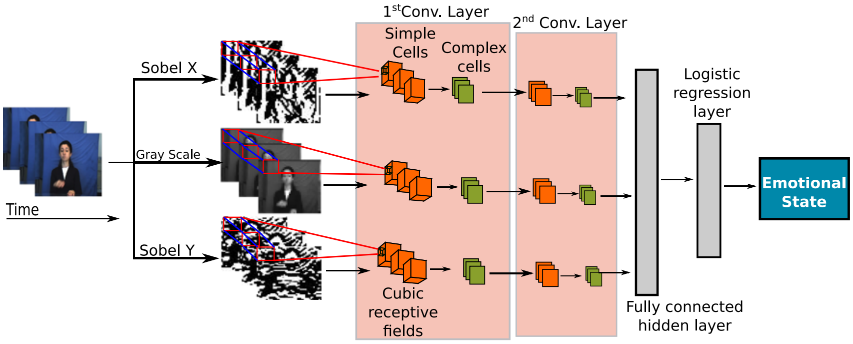


Fig. 3. Proposed architecture for a Multichannel Convolutional Neural Network using 3 channels. In the first layer, a cubic receptive field is implemented.

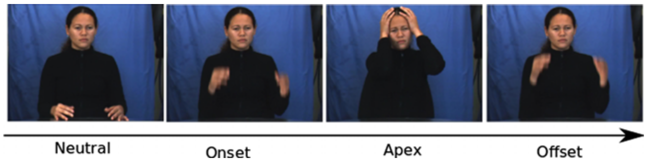


Fig. 4. Temporal phase evolution of the emotional state, starting with Neutral, Onset, Apex, Offset and ending with the return to Neutral.

model, with the applications of the three channels and the cubic receptive field.

2.4. Sequence dependency

In an emotion sequence, there are four temporal phases: onset, apex, offset, and neutral, as illustrated in Fig. 4 and discussed by Gunes and Piccardi (2009), who also state that the apex phase presents the most significant features for emotion determination. In this phase, the facial expression and motion of the subject are more intense and display unique body postures and face expressions within each emotion. Based on this fact, the authors apply a maximum-voting-of-apex-frames approach to identify the emotional state in a sequence.

Our model extracts the temporal dependency of the frames in a sequence and it is trained using a stack of frames which always has the same number of images. To train the network, we select only the frames from the apex phase of each emotion. The dataset we use is annotated, and the selection process is explained in Section 3. The frames present in the other temporal phases for all the emotions are collected and labeled as a neutral emotional state.

After training, the network is able to identify the emotion present in a sequence of frames and based on our experimental results, we obtained optimal results when using a small sequence of frames. The apex phase of each expression can vary from six to fifteen frames. The temporal features contained in a sequence of frames during the temporal phases are important and must be present in the sequence representation. Our model extracts spatio-temporal features from a sequence of 4 frames which are used as input to the network. To be able to deal with sequences with a larger number of frames, we create a sequence dependency scheme: A sequence of 4 frames is fed to the network, and a label is generated. To allow online processing, for each interval of 5 labels, which are obtained by the classification of 5 times 4 frames, a voting process is executed and the emotion with most labels is chosen. Note that these 20 frames underlying the voting will cover not only the apex phase, but also parts of the Onset and/or Offset phases.

To train the network, the frames from the other temporal phases, Neutral, Onset and Offset, are labeled as a neutral emotion and are also partitioned into smaller sequences with 4 frames.

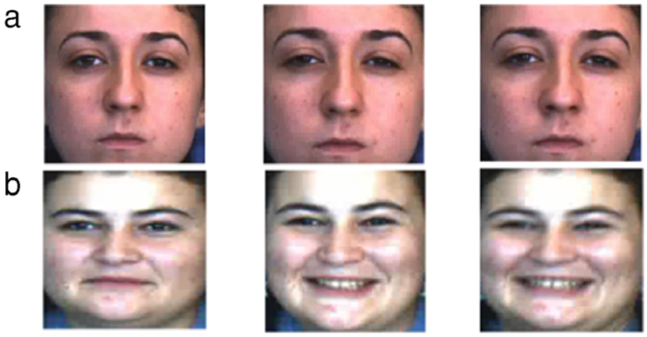


Fig. 5. Example sequences of apex phases from the face of the subject for (a) anger and (b) happiness.

Using this approach, we can feed our model with a live stream of frames, and it will be able to identify when an emotion is demonstrated, discerning different emotions by their apex phases. When none of the emotion expressions is recognized, our model will generate the "Neutral" label, which allows performing a temporal classification of emotions in an online classification scenario.

2.5. Facial expression recognition

To evaluate the contribution of the face on the emotional state recognition, only the face expression of the subject is used to train the model. The input structure is the same for all experiments, only the content of the images change.

The face is tracked using the Viola–Jones face detection algorithm, proposed in the work of Viola and Jones (2004), which uses an Adaboost-based detection. Wang (2014) discuss the Viola–Jones algorithm, and shows that it is robust and effective when applied in general face detection datasets, and even in real-world scenarios. In our experiments, the Viola–Jones algorithm was 100% correct and detected the positions and sizes of the faces in all presented images. After detection, the face is cropped from the image and used to construct the set of inputs for the network. Fig. 5 shows two examples of a sequence representing the apex phase of the faces from the subjects. Fig. 5(a) shows a sequence for anger and Fig. 5(b) shows a sequence for happiness.

Some of the sequences also contain head motion and are occluded by hands. These characteristics are also taken into consideration, and are not removed from the input, so that after training, the model can recognize occluded, partial and rotated faces which are present in the evaluated dataset.

2.6. Body motion recognition

For emotion expressions, especially spontaneous expressions, there is no pre-defined set of gestures. Each person demonstrates

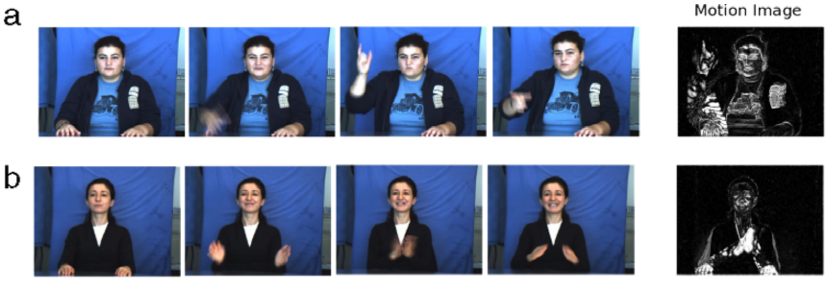


Fig. 6. Examples of motion image generated by all the frames in the apex phase from (a) anger and (b) happiness. We show only some examples of the apex phase for each of these emotions.

his/her own emotions using hands and arms and movements, which contain no structure and are highly dependent on the person's personality and attitude towards the situation, as discussed by Zhao and Badler (1998). To be able to use this input to classify emotions, we take into consideration the motion of the arms and body motion, and use a technique proposed in an early approach by Barros, Parisi, Jirak, and Wermter (2014), to model it.

To use the motion representation, an additional layer is added to the proposed network. This layer receives N gray scale frames, without the application of any additional preprocessing step, and creates a representation based on the difference of each pair of frames. The processing works in a sequential way, receiving the frames one after another. The layer computes an absolute difference and sums up the resulting frame to a stack of frames. This operation generates one image representing the motion of the sequence, defined here as M:

$$M = \sum_{i=1}^{N} |(F_{i-1} - F_i)| \, (i/t), \tag{4}$$

where N is the number of frames of the apex phase, F_i represents the current frame and (i/t) represents the weighted shadow. The weighted shadow is used to create different gray scale shadows in the final representation according to the time that each frame is presented. That means that every frame on the image will have a different gray tone in the final image. The weight t starts as 0 in the first frame and is increased over time, so each frame has a different weight. The absolute difference of each pair of frames removes non-changing parts of image, being able to remove the background or any other detail in the image that is not important for the motion representation. By summing up all the absolute differences of each pair of images it is possible to create a shape representation of the motion. This representation contains the shape of the motion and, with the help of the weighted shadows, the information of when each single posture happened.

Fig. 6 illustrates examples of motion image, M in Eq. (4), generated by all the frames of the apex phase of Fig. 6(a) anger and Fig. 6(b) happiness.

When using the motion image as input stimuli, our model does not apply a cubic receptive field, since the motion representation is acquired by the previous layer and not by the model itself. Each output of the network is already the final label for the full sequence used to generate the motion image.

2.7. Multimodal recognition

Here, the network is fed with the original frames, without the application of any pre-processing technique. The image sequence contains the upper-body part of the subject, having access to the face expression and body motion. Different from other approaches, this experiment does not perform manual feature fusion, but focuses on the learning process and the use of both, face and body information to create a unique representation. The representation extracted by the network while using the full sequence, is completely new and unique, without any correlation with the previous experiments. This is expected due to the nature of the MCCNN to adapt to the present stimuli and learn the most relevant features from the presented images. Examples of the sequences of the apex phases are shown in Fig. 7(a) for anger and in Fig. 7(b) for happiness.

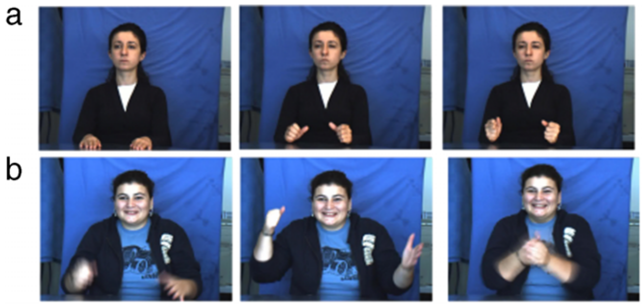


Fig. 7. Example of sequences of raw images from the apex phase of two emotions: (a) anger and (b) happiness.

Input images which our model uses as input are shown in Figs. 5–7. The model is the same, what changes are the input stimuli. The input images are presented always in gray scale and have the same resolution, and every sequence contains the same length.

3. Experiments

3.1. Methodology

To evaluate the proposed model on an emotional state recognition task, the bi-modal face and body benchmark database FABO, presented by Gunes and Piccardi (2006), is used. The database is composed of recordings of the face and body motion using two cameras, one of them capturing only the face and the second one capturing the upper body. Each video contains one subject executing the same expression in a cycle of two to four expressions per video.

The FABO dataset has annotations about the temporal phase of each video sequence. To create the annotation, six observers label each video independently and then a voting process is executed. We use only upper-body videos which have a voting majority regarding the temporal phase classification, similar to Gunes and Piccardi (2006). A total number of 281 videos are used and are shown in Table 1. The database contains ten emotional states: "Anger", "Anxiety", "Boredom", "Disgust", "Fear", "Happiness", "Surprise", "Puzzlement", "Sadness" and "Uncertainty". As necessary for the temporal labeling, only the apex state of each sequence is used for training the model. The other frames present in the remaining temporal phases are grouped into one new category named "Neutral" leading to a total of 11 emotional states to be classified.

Table 1
Number of videos available for each emotional state in the FABO dataset. Each video has 2–4 executions of the same expression.

Emotional state	Videos	Emotional state	Videos
Anger	60	Happiness	28
Anxiety	21	Puzzlement	46
Boredom	28	Sadness	16
Disgust	27	Surprise	13
Fear	22	Uncertainty	23

Table 2
Parameter sets evaluated for each experiment. The combination of all values for all parameters was evaluated and discussed.

Parameter	Values		
Filter maps layer 1 and 2	10 and 20	20 and 40	30 and 60
Receptive field size layer 1	3 × 3 × 2	11 × 11 × 2	21 × 21 × 2
Receptive field size layer 2	3 × 3	7 × 7	11 × 11

Table 3
Reported accuracy for each parameter combination computed during the parameter exploration experiments.

Receptive field size		Filter maps		
1st layer	2nd layer	10	20	30
3	3	**91.30%**	**91.25%**	**91.18%**
3	7	90.93%	89.83%	91.00%
3	11	89.93%	87.92%	90.47%
11	3	90.77%	90.83%	90.01%
11	7	90.04%	89.75%	91.02%
11	11	87.34%	88.65%	90.43%
21	3	90.08%	89.82%	89.90%
21	7	90.01%	88.92%	90.42%
21	11	87.89%	87.30%	88.90%

Three experiments are executed and evaluated. The first one uses information of the face expression to determine emotional estates. The second one extracts information from the body motion, composed by arms, torso and head movements, and the third one uses both types of information. Each experiment uses a different image type as input and has as purpose to evaluate which modality has the better representation for emotional state classification. Training and recognition time and accuracy results are collected. For the experiments, the images are resized to the same size: 100 pixels of height, keeping the same proportion of width. With images of the same size it is easier to compare how the information itself affects the training time of the network and the final classification results.

For all experiments, the network configuration remains the same: an MCCNN with 3 channels, one of them receiving the raw image and the other two the application of the Sobel filter. The network has two layers and a cubic receptive field is implemented on the first layer. The network receives two frames as input, to reduce the number of parameters to be updated during training. In our experiments, two frames were found to be enough to discriminate the apex phase from the others. Every sequence of 3 labeled outputs is used in the voting scheme to classify the emotional state. This makes 6 frames necessary to determine an emotional state. For all experiments, three-fold cross validation is performed, and the average performance over 30 trials is reported in this work. Each sequence used as input for the model, during training and testing, is composed of subsequent images of the same subject. The temporal phases are given by the dataset, and the sequences are generated based on them. For the face expression and multimodal experiments all sequences have the same length. In the body motion experiment, the input is composed by full sequence of frames of the apex phase, so there is no necessity of a voting scheme. The output of the network is used as final emotion classification.

To compare our network with the common CNN implementation, we also evaluate each channel individually. Each channel is implemented as a common CNN, using the cubic receptive field in the first layer, and the same parameters as the other experiments. The accuracy and standard deviation are collected and compared to the MCCNN results.

To evaluate how the parameters affect the network, a parameter exploration experiment is performed. Three parameters are chosen, with the range of values based on the suggestions of Simard, Steinkraus, and Platt (2003) and our previous experiments with CNNs. The three parameters were chosen because of their major influence on the network response. A total of three different values are chosen for each parameter, generating a total of 27 experiments. Table 2 shows the chosen parameters and the range of values.

The number of filter maps affects directly the amount of features extracted, and what these features represent. A large number of feature maps introduces redundancy, and a small number is not enough to extract a proper description of the emotion sequence. The minimum and maximum values of 10 and 30 filter maps were chosen based on preliminary experiments, where these values represented the limits where the network showed a big variation for the accuracy. The number of filter maps on the second layer, as suggested by Simard et al. (2003), is selected as twice the number of filter maps on the first layer. This selection leads to more specialized features on the second layer to expand the representations on the first layer, which are mostly edge-like detectors. The size of the receptive fields determines which pixel structures are important for the model. On the first layer, the receptive fields are connected directly to the image, and they will enhance structures present in the original data. If the receptive fields are too small, they will not be able to enhance important pixel structures, and will generate redundancy for the next layers. If they are too large, they will absorb more pixel structures than necessary, and they will not be able to determine or to react to these structures, aggregating more than one structure into one filter map. This could generate very specific filter maps for the data while training the network, which leads to an overfitting of the model. For our experiments, we chose a range between the smaller and maximum receptive field sizes which were able to extract meaningful information from our input.

For each parameter set, 30 experiments are performed and the averages of the accuracy results are reported in this work. The effect of the selected parameters on the trained data is evaluated and discussed in the next sections of this work.

3.2. *Parameters evaluation*

After performing the parameter exploration experiments, the average of the accuracy was computed. Table 3 shows the results for all the parameter combinations. For the first set of experiments, we locked the number of filter maps in 10. The best result was achieved with a configuration of a receptive field size in the first and second layer of 3 × 3 pixels, with an accuracy of 91.3%, while the worst result found, with a configuration of kernel size in the first layer of 11 × 11 pixels and in the second layer of 21 × 21 pixels, was 87.89%. We could find a trend: when the size of the receptive fields was increased, in both layers, the network produced the poorer results.

When locking the number of filter maps on the first layer at 20, the results obtained showed a similar trend: increasing the size of the receptive fields of the filter maps for the first and second layer decreases the accuracy. For this set of experiments, the best result was also with the smaller receptive field size, in both layers, achieving 91.25% of accuracy. The worst result can be observed

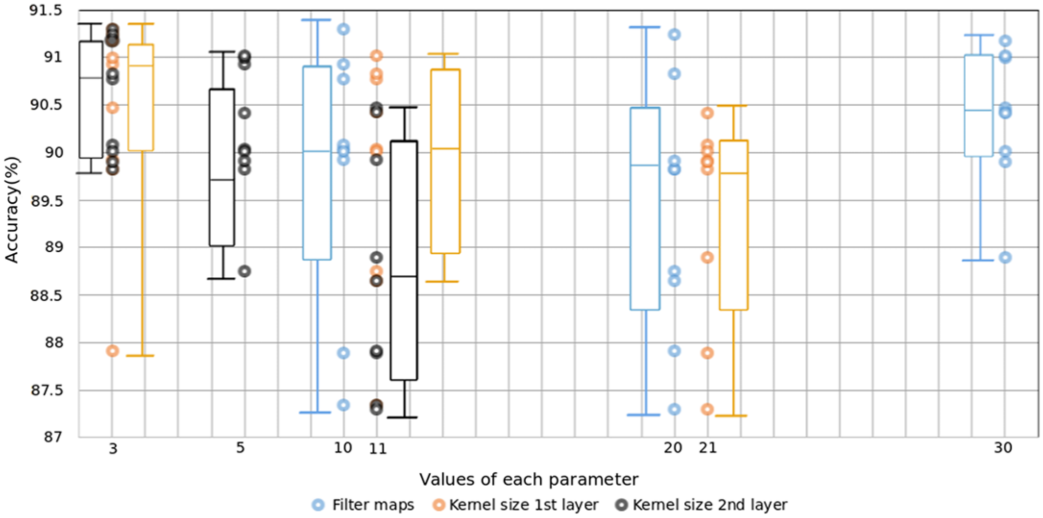


Fig. 8. Individual analysis for the parameter exploration. It is possible to see a trend that when the kernel sizes are smaller the accuracy tends to be higher. When increasing the number of filter maps, the average of the results is also higher, despite the best accuracy being lower.

when using the maximum value of the receptive field size. This configuration achieved an accuracy of 87.3%.

The trend can also be found when the number of filter maps on the first layer was locked at 30. The best result was also with the smaller receptive field sizes with an accuracy of 91.18%. The worst accuracy was 88.9%, when using the largest kernel size for both layers.

Evaluating the parameters, it is possible to find some trends in the network behavior. Fig. 8 shows a box plot with the individual analysis of the parameters. The plot shows the variance of the accuracy of each parameter. The plot depicts that using a smaller receptive field in both layers the accuracy improves. Looking at the plot, it is possible to see a clear trend in the spread of the results. When using smaller receptive fields, the network produces results with a better accuracy median and smaller variance. Also, increasing the number of filter maps decreases the accuracy variance between the experiments. This shows that when we increase the number of filter maps, the results of our network tend to be more stable. Using a smaller receptive field allows the network to learn those general structures which occur more often in the images. Passing these general features through our layers generates higher level features in deeper layers. But, as our network is not very deep, we need to increase the number of filter maps in order to expand the variety of features extracted in the same layer.

When evaluating the combinations of parameters, it is possible to visualize how the number of filter maps influence the results. Fig. 9 illustrates the plot with the combination of the parameters. It is possible to see that when we increased the size of filter maps, the variance of the accuracies is small. Also, is possible to see that increasing the number of filter maps in the second layer produces a lower accuracy. However, when using 30 filter maps and using a receptive field of 7 × 7 in the second layer, the network produces better results. This occurs because when extending the number of filter maps, the network generates different feature representation at the same level and thus generates redundant features which allow a better generalization. It is important to note that with too many redundant filters, the network lose the capability of generalizing and ends up overfitting.

Analyzing the model after training, the first layers of the MCCNN were able to act as a low-level feature extractor that, when comprised in the last layers, was able to extract features which represent the face expression. Fig. 10 shows an example of the feature representation of a sequence passing through some of the filters in the network. It is possible to see the inner representations of the network. For example, in the first layer it is possible to see that what the filters extract are regions like eyes, mouth and eyebrows. After the first layer it is difficult to interpret the representation, but as each layer receives the representation from a previous layer, the network encodes hierarchical representation.

Table 4
Average accuracy for all the experiments: *Facial expression*, *Body motion* and *Multimodal*.

Experiment	Accuracy (%)
Facial expression	72.70% (±3.1)
Body motion	57.84% (±7.7)
Multimodal	91.30% (±2.7)

For the final experiments, the parameters chosen were 10 filter maps for the first layer and a receptive field of 3 × 3 pixels implemented in both layers. Although the accuracy is similar when using any of the three filter map numbers, the training time when using 10 filter maps is smaller. When using 10 filter maps an average of 4327.88 s (1 h 12 min) was necessary for training which was faster than using 20, with an average of 7253.74 s (2 h), and 30 filter maps with an average of 12 873.64 s (3 h 30 min). All experiments were performed on a desktop machine with an Intel Xeon E5630 processor, an Nvidia GeForce GTX 590 graphics cards and 24 GB of RAM.

3.3. *Results*

The averages of accuracy, training time and recognition obtained for all three experiments are shown in Table 4. As expected, the *Body Motion* experiment delivered the worst result, with an accuracy of 57.84%. As no structured motion is present, the body movements act like a complement for facial expression. Because of the representation of motion in one single frame and the use of a common receptive field instead of a cubic one, the motion experiment has less input stimuli and less parameters to train. Therefore the training time was almost 10 times less than for the other experiments, as reported in Table 5.

A total of 100 training epochs were performed for each experiment, which was shown to be enough to achieve convergence. Table 5 shows the accuracy of each experiment on the 20th epoch, and the *Facial Expression* had a higher accuracy than the *Multimodal* experiment. In the *Facial Expression* experiment only the face was present, which leads to faster convergence because the variations

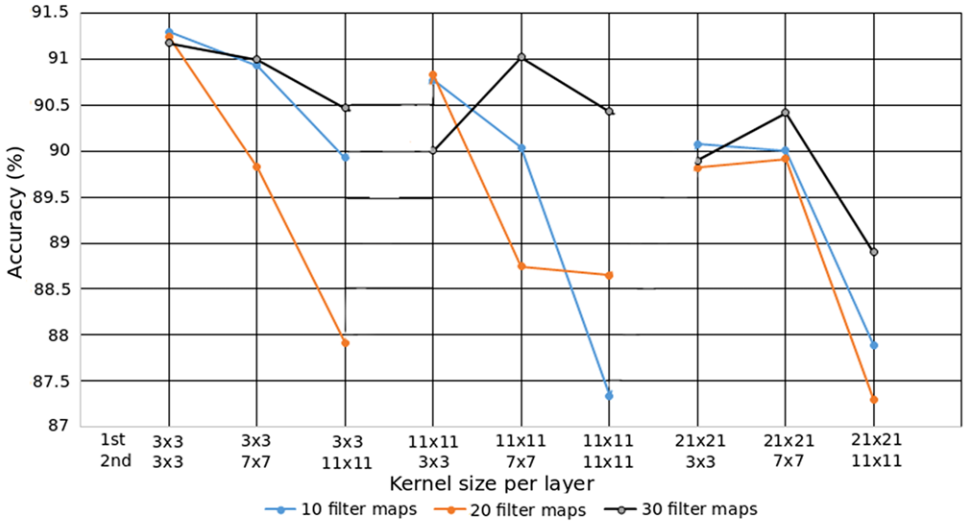


Fig. 9. Combination analysis for the parameter exploration. When smaller receptive fields are used, the accuracy is higher. Also, when comparing the same size of receptive fields, but increasing the number of filter maps, it is possible to see that the average of the accuracy increases, although the best result was found with the smaller number of filter maps.

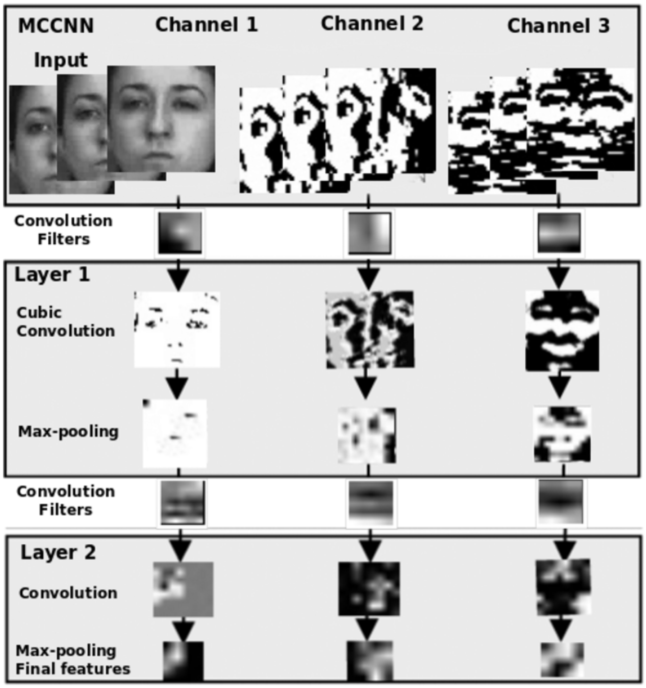


Fig. 10. Example of the output of the network for some filters in each layer of the network. The first channel extracts more complex structures than the other two, as regions with eyes or mouth.

Table 5
Average of training time and accuracy on the 20th epoch for all experiments.

Experiment	Training times (s)	Accuracy 20th epoch
Facial expression	4327 (1 h 12 min)	60.80%
Body motion	412 (7 min)	40.36%
Multimodal	4455 (1 h 13 min)	52.75%

within each class rely on the face expression itself, and not on different subjects, clothes or any other part of the image. In the *Multimodal* experiment the model received an image containing not only the face of the person but also with the background. When the whole subject changes and a larger part of the image is analyzed, the filters tend to take more time until they can learn that the structures of the face are important for the final classification, and not the changes of the subject's clothes or gender, for example.

Table 6
Comparison of the state-of-the-art models for the Facial expression experiment.

Approach	Accuracy (%)
CNN—gray scale	63.36%
CNN—Sobel X	58.20%
CNN—Sobel Y	57.80%
MCCNN	**72.70%**
Temporal normalization (Chen et al., 2013)	66.50%
Bag of words (Chen et al., 2013)	59.00%
SVM (Gunes & Piccardi, 2009)	32.49%
Adaboost (Gunes & Piccardi, 2009)	35.22%

For the *Facial Expression* experiment, an average of 72.70% of accuracy was obtained. Our model was able to learn more information from facial expressions than from motion only. When the multimodal representation was used, the model achieved an average accuracy of 91.30%. The recognition time for all three experiments was very similar, around 6 ms, due to using the same network topology and input image size. The training time was similar for the *Facial Expression* and *Multimodal* experiments, because both experiments used the same number of images.

Our model achieves higher accuracy for the experiments with *Face expression* and *Multimodal* experiments, compared with the results reported in Chen et al. (2013). Table 6 shows the results for the *Face expression* experiment, also compared to the work of Gunes and Piccardi (2009). Chen et al. (2013) report the results using a video-based approach. The results reported by Gunes and Piccardi (2009) for Adaboost and SVM are collected using a frame-based accuracy. The proposed model improves the accuracy by more than 6% compared to Chen et al. (2013). Compared to the common CNN architecture, our model improved the accuracy in almost 10%. We evaluated the three channels individually, and the channel receiving the gray scale images was the one with higher accuracy, 63.36%, against 58.2% and 57.8% obtained by the Sobel X and Y channels respectively. This shows that the CNN has competitive results, when compared to the models of Chen et al. (2013) and Gunes and Piccardi (2009), and when using the MCCNN implementation, it was possible to outperform these models.

The work of Chen et al. (2013) compares different feature extractions for the *Body motion* experiment. Their reported accuracy

Table 7
Comparison of the state-of-the-art models for the Body motion experiment.

Approach	Accuracy (%)
CNN—gray scale	53.32%
CNN—Sobel X	51.35%
CNN—Sobel Y	51.85%
MCCNN	57.84%
Temporal normalization (Chen et al., 2013)	66.70%
Bag of words (Chen et al., 2013)	65.30%
SVM (Gunes & Piccardi, 2009)	64.51%
Random forest (Gunes & Piccardi, 2009)	**76.00**%

Table 8
Comparison of the state-of-the-art models for the *Multimodal* experiment.

Approach	Accuracy (%)
CNN—gray scale	74.5%
CNN—Sobel X	69.7%
CNN—Sobel Y	71.2%
MCCNN	**91.3**%
Chen et al. (2013)	75.0%
Gunes and Piccardi (2009)	82.5%

is almost 9% greater than that of our model, but the feature selection is more complex and also depends on explicit feature tracking, like hands and face position. Also, they tracked several face components, hands and shoulders. Gunes and Piccardi (2009) use a complex set of features to describe motion, composed of optical flow, edginess, geometry features among others. Although the results of our model are not as good as in these experiments, there are no pre-processing steps on the images and thus less computational processing is necessary. Table 7 shows the comparisons of the accuracy. The common CNN architecture also achieved a lower accuracy of 53.32% when using the gray scale image, and 51.35% and 51.85% when using the Sobel in X and Y directions respectively. This shows that the MCCNN implementation does not improve much the recognition of motion images. Also, all the channels, individually, obtained a similar accuracy, indicating that the application of Sobel filters in the image does not improves the recognition of motion images.

In the work of Chen et al. (2013), and also in the work of Gunes and Piccardi (2009), the multimodal recognition, containing information of face expression and body motion, is achieved by manual fusion of features. Table 8 shows the comparison between their results and those obtained with our model. Gunes and Piccardi (2009) have a set of complex features extracted for face and body motion. Classifying these feature vectors with an Adaboost with random forest classifier achieved the best accuracy of 82.65%. Chen et al. (2013) use a similar approach and integrate all their features using a fusion-based approach and achieve an accuracy of more than 75% when using the bag-of-words method for temporal segmentation. Our model achieves a total of 91.3% accuracy, almost 10% more than the one reported by Gunes and Piccardi (2009), without using any explicit feature selection. Using the common CNN implementation, the results were lower than all the others. This was due to the fact that each separated channel separated did not have enough power to learn and extract meaningful features from the upper body. The channels which received the Sobel operator were the ones with the lower accuracy, 69.7% and 71.2% for X and Y respectively. This is explained by the fact that for the upper-body images the face is not the most prominent region in the image and when the Sobel filters are applied, the image loses some of the small structures that composes the face. When using the MCCNN architecture, this effect does not happen since the channels are able to extract information from the original image and the one which the Sobel filter was applied to.

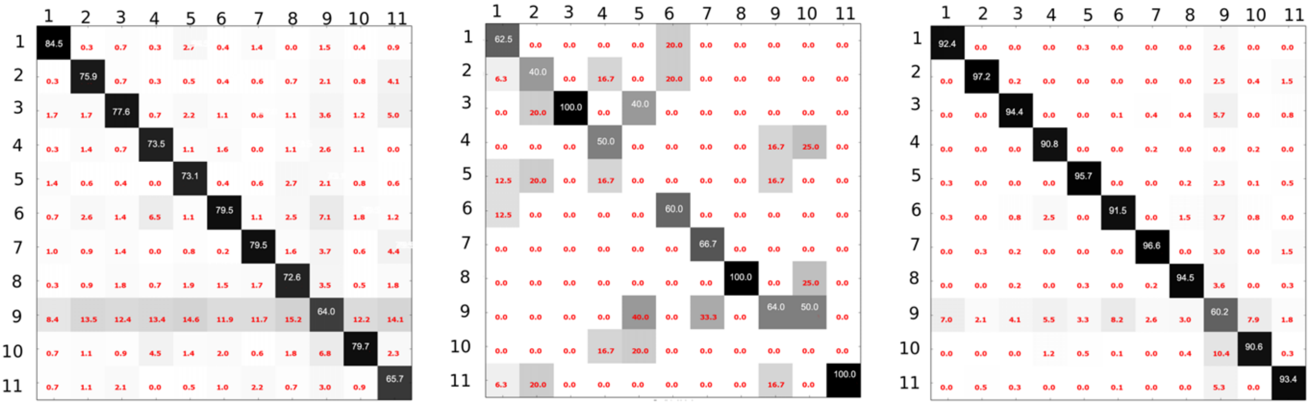
Fig. 11 shows the confusion matrices for our three experiments. For the *Facial Expression* and *Multimodal* experiments, we can see that the class which is most difficult to classify correctly is the Neutral class. This class has the largest number of sequences and contains examples of every subject in the dataset, from all temporal phases except the apex phase, which explains why it is the most misclassified class. We can also see that the other classes have good recognition rates, as discussed before. In the *Facial Expression* experiment, the model was able to create a clear distinction for each emotion expression, having only a few misclassifications when trying to distinguish between Boredom, Anxiety, Puzzlement and Uncertainty. This happens, because these classes have similar face expressions. Looking at the *Body Motion* experiment, we see that the model creates a better distinction for these classes, and the model was able to classify all the examples of Uncertainty and Boredom correctly. On the other hand, most of the other expression classes failed, indicating that motion alone is not enough to classify these expressions. The *Multimodal* experiment has the best results, as can be seen in its confusion matrix. Still the Neutral class is the one with most misclassifications, for the same reason as in the other experiments. The model still shows some misclassification when trying to classify Uncertainty and Anxiety, but with a smaller percentage than in the *Facial Expression* experiment. A deeper future analysis on the emotion expression itself would help to understand better the generalization capabilities of the model.

4. Discussion

The proposed Multichannel Convolutional Neural Network (MCCNN) architecture extends the concept of a convolutional neural network by using more than one channel, two of them receiving specialized information from encoded edge enhancement layers, to be able to deal with multimodal information, in this case facial expression and body posture.

Based on the parameter exploration experiments, the results show that the MCCNN performs better when using smaller receptive fields on both layers, extracting information from smaller patches of the image. When applying more than one channel, more information is extracted from the images. As the two channels receive the images from the Sobel filter layers, the remaining one will be influenced, and will specialize on more complex features. This will train the small filter maps on the first layers to find complex patterns in the image and not only edges. Increasing the number of filter maps will generate more different filters, which will increase the number of features extracted from the image. The drawback of increasing the number of filter maps is an increase of the connections and the parameters to be updated during training, which could lead to overfitting.

Our parameter evaluation was done based on the FABO dataset. Each subject moves in front of the camera in different ways, with different directions, velocity and body/face postures and expressions. The subjects themselves wear different clothes, accessories (glasses, watches, ear rings, among others) and have different gender, age and ethnicity. With the appropriate datasets, adapting the network topology and tuning the parameters will allow the network to be robust for some problems such as: the distance of the subject from the camera, by adding more simple/complex layers; the presence of more than one subject, by slicing the image into sections to be processed by the network once per time; and occlusion of the subject in the middle of the sequence, by increasing the number of filter maps in each layer. Also a further analysis of the deep layers of the network could reveal which kind of complex features are learned during training. This analysis could help to understand how these features are influencing each of the channels, and clarify how strong the influence of the Sobel-based channels is on the third channel.



(a) Facial Expression experiment.

	1	2	3	4	5	6	7	8	9	10	11
1	84.5	0.3	0.7	0.3	2.7	0.4	1.4	0.0	1.5	0.4	0.9
2	0.3	75.9	0.7	0.3	0.5	0.4	0.6	0.7	2.1	0.8	4.1
3	1.7	1.7	77.6	0.7	2.2	1.1	0.6	1.1	3.6	1.2	5.0
4	0.3	1.4	0.7	73.5	1.1	1.6	0.0	1.1	2.6	1.1	0.0
5	1.4	0.6	0.4	0.0	73.1	0.4	0.6	2.7	2.1	0.8	0.6
6	0.7	2.6	1.4	6.5	1.1	79.5	1.1	2.5	7.1	1.8	1.2
7	1.0	0.9	1.4	0.0	0.8	0.2	79.5	1.6	3.7	0.6	4.4
8	0.3	0.9	1.8	0.7	1.9	1.5	1.7	72.6	3.5	0.5	1.8
9	8.4	13.5	12.4	13.4	14.6	11.9	11.7	15.2	64.0	12.2	14.1
10	0.7	1.1	0.9	4.5	1.4	2.0	0.6	1.8	6.8	79.7	2.3
11	0.7	1.1	2.1	0.0	0.5	1.0	2.2	0.7	3.0	0.9	65.7

(b) Body Motion experiment.

	1	2	3	4	5	6	7	8	9	10	11
1	62.5	0.0	0.0	0.0	0.0	20.0	0.0	0.0	0.0	0.0	0.0
2	6.3	40.0	0.0	16.7	0.0	20.0	0.0	0.0	0.0	0.0	0.0
3	0.0	20.0	100.0	0.0	40.0	0.0	0.0	0.0	0.0	0.0	0.0
4	0.0	0.0	0.0	50.0	0.0	0.0	0.0	0.0	16.7	25.0	0.0
5	12.5	20.0	0.0	16.7	0.0	0.0	0.0	0.0	16.7	0.0	0.0
6	12.5	0.0	0.0	0.0	0.0	60.0	0.0	0.0	0.0	0.0	0.0
7	0.0	0.0	0.0	0.0	0.0	0.0	66.7	0.0	0.0	0.0	0.0
8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100.0	0.0	25.0	0.0
9	0.0	0.0	0.0	0.0	40.0	0.0	33.3	0.0	64.0	50.0	0.0
10	0.0	0.0	0.0	16.7	20.0	0.0	0.0	0.0	0.0	0.0	0.0
11	6.3	20.0	0.0	0.0	0.0	0.0	0.0	0.0	16.7	0.0	100.0

(c) Multimodal experiment.

	1	2	3	4	5	6	7	8	9	10	11
1	92.4	0.0	0.0	0.0	0.3	0.0	0.0	0.0	2.6	0.0	0.0
2	0.0	97.2	0.2	0.0	0.0	0.0	0.0	0.0	2.5	0.4	1.5
3	0.0	0.0	94.4	0.0	0.0	0.1	0.4	0.4	5.7	0.0	0.8
4	0.0	0.0	0.0	90.8	0.0	0.0	0.2	0.0	0.9	0.2	0.0
5	0.3	0.0	0.0	0.0	95.7	0.0	0.0	0.2	2.3	0.1	0.5
6	0.3	0.0	0.8	2.5	0.0	91.5	0.0	1.5	3.7	0.8	0.0
7	0.0	0.3	0.2	0.0	0.0	0.0	96.6	0.0	3.0	0.0	1.5
8	0.0	0.0	0.2	0.0	0.3	0.0	0.2	94.5	3.6	0.0	0.3
9	7.0	2.1	4.1	5.5	3.3	8.2	2.6	3.0	60.2	7.9	1.8
10	0.0	0.0	0.0	1.2	0.5	0.1	0.0	0.4	10.4	90.6	0.3
11	0.0	0.5	0.3	0.0	0.0	0.1	0.0	0.0	5.3	0.0	93.4

Fig. 11. Confusion matrices for the (a) *Facial Expression* experiment, (b) the *Body Motion* experiment and (c) the *Multimodal* experiment. Each column and row represent one emotion expression: 1—Anger, 2—Anxiety, 3—Boredom, 4—Disgust, 5—Fear, 6—Happiness, 7—Puzzlement, 8—Sadness, 9—Neutral, 10—Surprise, 11—Uncertainty. The columns show the true label, and the rows the predicted label. The numbers represent the percentages of predicted labels. The numbers on the diagonal are correct predictions.

Our model was evaluated using the same methodology for experiments established by Gunes and Piccardi (2009). Two approaches using the same methodology were selected for comparison, the work of Chen et al. (2013) and Gunes and Piccardi (2009). To evaluate the *Face expression,Gunes2009* extract a series of features from the face. They first track eyes, eyebrows, mouth, nose, chin, and other face regions. For each region, they apply a series of 150 different feature descriptors. This process consumes a lot of computational power for feature extraction. In the research of Chen et al. (2013), they use 53 face landmarks to extract some features and classify them with an SVM. Our model does not use specific predefined features. The images are presented to the network, and the model learns which are the most relevant features in the image for the emotional state classification task. This process reduces the computational power necessary for training and recognition, and eliminates the constraints of using specific features, which could be listed in this experiment as illumination change, different skin color tones and occlusion among others. Our model was measured with a higher accuracy when compared with the approaches of Chen et al. (2013) and Gunes and Piccardi (2009), showing that our feature learning is more reliable.

For the *Body Motion* experiment, our model provided worse results than the ones reported in the studies of Chen et al. (2013) and Gunes and Piccardi (2009). This could be due to the fact that to be used in a gesture recognition task, the proposed model needs a more structured motion representation. The work of Gunes and Piccardi (2009) creates a motion representation tracking different parts of the human: facial land marks, shoulder and hands. Their representation is obtained by the application of 144 different feature extraction techniques in each body part. In the work of Chen et al. (2013) they apply skin-color segmentation, head and hand tracking techniques to create a segmented image. After that, they generate motion history images to represent the body movement.

In the *Multimodal* experiment, our model presented the best results. In both the studies of Chen et al. (2013) and Gunes and Piccardi (2009), a manual feature fusion is implemented. This manual feature fusion approach showed good results, but only if the feature representation for both body motion and facial expression were well defined. The increase in the number of features to be classified has to be taken into consideration, which could lead to redundant or concurrent features. This approach relies on different feature extraction techniques, each one of them having its own constraints. Our model uses a different approach. As the feature extraction is adapted to the presented images, the MCCNN will learn a unique feature representation based on the information in the image. When a sequence is presented to the network, the network will learn to extract motion and face features from the full sequence. It generates a new feature vector that is not composed of the elements of the previous experiments. Our strategy presented best results, and increased the accuracy by almost 10 % compared to Gunes and Piccardi (2009) and more than 15 % in comparison with Chen et al. (2013).

Our model was also compared with the CNN and for all the experiments our model outperformed it. We evaluated each of our channels individually, and we realized that each channel alone did not extract enough information. In particular, where the face expression is only one part of the image, a common CNN could not learn meaningful information. For the body motion experiment, we showed that a common CNN is enough to learn information from a motion image, and the application of the Sobel filter is not necessary for this case.

Using multimodal information for emotion recognition was shown to be the best option. In the studies of Chen et al. (2013) and Gunes and Piccardi (2009), and also ours, the combination of modalities achieved the highest accuracy in an automatic emotional state recognition task. This result agrees with psychological and social studies of Gu et al. (2013) and Kret et al. (2013). The proposed model improves the state-of-the-art research by using a hierarchical feature representation, capable of learning different features to be extracted depending on the presented stimuli. The features were learned based on the input type, when the face of a person is shown. The first layers are able to extract complex shapes, like eyes, mouths and eyebrows, which are used for the deeper layers to build more complex representation. To know exactly what kind of features the network learns, is part of our future work and will need deeper analysis with appropriate techniques, such as the method of Zeiler and Fergus (2014) for the visualization of deep neural network features.

5. Conclusion and future works

We proposed a novel implementation of convolutional neural networks, called Multichannel Convolutional Neural Network (MCCNN), applied to multimodal automatic emotional state recognition. The model is able to deal with sequential frames, for multimodal visual stimuli classification.

The network is evaluated with an established dataset in three different experiments. Each experiment evaluates how different visual modalities contribute to emotional state recognition. Consistent with findings in the literature on psychological and social studies, the proposed model achieves a better performance when multimodal information is used, in this case composed of face expression and body motion.

The MCCNN architecture is shown to achieve better results than the state-of-the-art approaches. The proposed model is able to learn unique feature representations from three different information streams: motion, face expression and both streams. The model can recognize a sequence of emotions from different subjects, expressed spontaneously, without any kind of previously determined structure, and in different positions from the camera. Also, the representations of the emotions learned by the proposed model are independent of specific feature extraction techniques, and independent of background and foreground segmentation. This indicates that our approach has a promising applicability for indoor human–robot interaction scenarios.

For future work, a further analysis of the features learned by the architecture in each experiment would help to visualize the complex features extracted by the model. Implementation of the model in a real-world scenario will be explored, to extend the model to real-time continuous recognition.

Acknowledgments

The authors thank Dr. Hatice Gunes for providing the FABO Database. This work was partially supported by CAPES Brazilian Federal Agency for the Support and Evaluation of Graduate Education (p.n.5951–13–5).

The authors also would like to thank Katja Koesters and Johannes Bauer for their constructive comments and insightful suggestions that improved the quality of this manuscript.

References

Adolphs, R. (2002). Neural systems for recognizing emotion. *Current Opinion in Neurobiology*, *12*(2), 169–177.

Aggarwal, J., & Ryoo, M. (2011). Human activity analysis: A review. *ACM Computing Surveys*, *43*(3), 16:1–16:43.

Bar, M. (2007). The proactive brain: Using analogies and associations to generate predictions. *Trends in Cognitive Sciences*, *11*(7), 280–289.

Barros, P., Parisi, G., Jirak, D., Wermter, S., et al. (2014). Real-time gesture recognition using a humanoid robot with a deep neural architecture. In *2014 14th IEEE-RAS international conference on humanoid robots (Humanoids)* (pp. 646–651). IEEE.

Breazeal, C., & Brooks, R. (2004). Robot emotions: A functional perspective. In J. Fellous (Ed.), *Who needs emotions*. Oxford University Press.

Cabanac, M. (2002). What is emotion? *Behavioural Processes*, *60*(2), 69–83.

Castellano, G., Kessous, L., & Caridakis, G. (2008). Emotion recognition through multiple modalities: Face, body gesture, speech. In C. Peter, & R. Beale (Eds.), *Lecture notes in computer science*: *Vol. 4868*. *Affect and emotion in human–computer interaction* (pp. 92–103). Berlin, Heidelberg: Springer.

Chen, S., Tian, Y., Liu, Q., & Metaxas, D. N. (2013). Recognizing expressions from face and body gesture by temporal normalized motion and appearance features. *Image and Vision Computing*, *31*(2), 175–185. affect Analysis In Continuous Input.

Ekman, P., & Friesen, W. V. (1971). Constants across cultures in the face and emotion. *Journal of Personality and Social Psychology*, *17*(2), 124–129.

Ekman, P., & Friesen, W. V. (1978). *The facial action coding system*. Consulting Psychologists Press Inc..

Erhan, D., Bengio, Y., Courville, A., Manzagol, P.-A., Vincent, P., & Bengio, S. (2010). Why does unsupervised pre-training help deep learning? *Journal of Machine Learning Research*, *11*, 625–660.

Fasel, B. (2002). Head-pose invariant facial expression recognition using convolutional neural networks. In *Proceedings of the fourth IEEE international conference on multimodal interfaces, 2002* (pp. 529–534).

Fukushima, K. (1980). Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, *36*(4), 193–202.

Gadanho, S. C. (2003). Learning behavior-selection by emotions and cognition in a multi-goal robot task. *Journal of Machine Learning Research*, *4*, 385–412.

Glorot, X., Bordes, A., & Bengio, Y. (2011). Deep sparse rectifier neural networks. In *Gordon, G. J., Dunson, D. B. (Eds.), Proceedings of the fourteenth international conference on artificial intelligence and statistics (AISTATS-11). Vol. 15. Journal of machine learning research—workshop and conference proceedings* (pp. 315–323).

Gu, Y., Mai, X., & Luo, Y.-j. (2013). Do bodily expressions compete with facial expressions? Time course of integration of emotional signals from the face and the body. *PLoS One*, *8*(7), 736–762. 07.

Gunes, H., & Piccardi, M. (2006) A bimodal face and body gesture database for automatic analysis of human nonverbal affective behavior. In *18th International conference on pattern recognition, ICPR, 2006, Vol. 1* (pp. 1148–1153).

Gunes, H., & Piccardi, M. (2009). Automatic temporal segment detection and affect recognition from face and body display. *IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics*, *39*(1), 64–84.

Hinton, G. E., Osindero, S., & Teh, Y.-W. (2006). A fast learning algorithm for deep belief nets. *Neural Computing*, *18*(7), 1527–1554.

Hubel, D. H., & Wiesel, T. N. (1959). Receptive fields of single neurons in the cat's striate cortex. *Journal of Physiology*, *148*, 574–591.

Ji, S., Xu, W., Yang, M., & Yu, K. (2013). 3D convolutional neural networks for human action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *35*(1), 221–231.

Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., & Fei-Fei, L. (2014). Large-scale video classification with convolutional neural networks. In *2014 IEEE conference on computer vision and pattern recognition, CVPR* (pp. 1725–1732).

Kret, M. E., Roelofs, K., Stekelenburg, J., & de Gelder, B. (2013). Salient cues from faces, bodies and scenes influence observers face expressions, fixations and pupil size. *Frontiers in Human Neuroscience*, *7*, 810–850.

Le, Q. V., Zou, W. Y., Yeung, S. Y., & Ng, A. Y. (2011). Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis. In *Proceedings of the 2011 IEEE conference on computer vision and pattern recognition*, CVPR'11. (pp. 3361–3368). Washington, DC, USA: IEEE Computer Society.

Lecun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 2278–2324.

Lerner, J. S., & Keltner, D. (2000). Beyond valence: Toward a model of emotion-specific influences on judgement and choice. *Cognition and Emotion*, *14*(4), 473–493.

Morishima, S., & Harashima, H. (1993). Facial expression synthesis based on natural voice for virtual face-to-face communication with machine. In *Virtual reality annual international symposium, 1993., 1993 IEEE* (pp. 486–491).

Ramirez-Amaro, K., Kim, E.-S., Kim, J., Zhang, B.-T., Beetz, M., & Cheng, G. (2013). Enhancing human action recognition through spatio-temporal feature learning and semantic rules. In *2013 13th IEEE-RAS international conference on humanoid robots, Humanoids* (pp. 456–461).

Ranzato, M., Huang, F.J., Boureau, Y.-L., & LeCun, Y. (2007). Unsupervised learning of invariant feature hierarchies with applications to object recognition. In *IEEE conference on computer vision and pattern recognition, CVPR* (pp. 1–8).

Simard, P., Steinkraus, D., & Platt, J. C. (2003). Best practices for convolutional neural networks applied to visual document analysis. In *International conference on document analysis and recognition* (pp. 958–963).

Spexard, T., Hanheide, M., & Sagerer, G. (2007). Human-oriented interaction with an anthropomorphic robot. *IEEE Transactions on Robotics*, *23*(5), 852–862.

Tokuno, S., Tsumatori, G., Shono, S., Takei, E., Suzuki, G., Yamamoto, T., & Shimura, M. (2011). Usage of emotion recognition in military health care. In *Defense science research conference and expo, DSR, 2011* (pp. 1–5).

Velusamy, S., Kannan, H., Anand, B., Sharma, A., & Navathe, B. (2011). A method to infer emotions from facial action units. In *2011 IEEE international conference on acoustics, speech and signal processing, ICASSP* (pp. 2028–2031).

Viola, P., & Jones, M. (2004). Robust real-time face detection. *International Journal of Computer Vision*, *57*(2), 137–154.

Wagner, J., Andre, E., Lingenfelser, F., & Kim, J. (2011). Exploring fusion methods for multimodal emotion recognition with missing data. *IEEE Transactions on Affective Computing*, *2*(4), 206–218.

Wallis, G., Rolls, E., & Földiák, P. (1993). Learning invariant responses to the natural transformations of objects. In *International joint conference on neural networks* (pp. 1087–1090).

Wang, Y.-Q. (2014). An analysis of the Viola-Jones face detection algorithm. *Image Processing On Line*, *4*, 128–148.

Wang, L., Liu, T., Wang, G., Chan, K. L., & Yang, Q. (2015). Video tracking using learned hierarchical features. *IEEE Transactions on Image Processing*, *24*(4), 1424–1435.

Wiskott, L., & Sejnowski, T. J. (2002). Slow feature analysis: Unsupervised learning of invariances. *Neural Computing*, *14*(4), 715–770.

Zeiler, M., & Fergus, R. (2014). Visualizing and understanding convolutional networks. In D. Fleet, T. Pajdla, B. Schiele, & T. Tuytelaars (Eds.), *Lecture notes in computer science*: *Vol. 8689*. *Computer vision—ECCV 2014* (pp. 818–833). Springer International Publishing.

Zhao, L., & Badler, N. (1998). Gesticulation behaviors for virtual humans. In *Sixth pacific conference on computer graphics and applications, 1998* (pp. 161–168).