Face Detection Using Convolutional Neural Networks and Gabor Filters

Bogdan Kwolek

Rzeszów University of Technology
W. Pola 2, 35-959 Rzeszów, Poland
bkwolek@prz.rzeszow.pl

Abstract. This paper proposes a method for detecting facial regions by combining a Gabor filter and a convolutional neural network. The first stage uses the Gabor filter which extracts intrinsic facial features. As a result of this transformation we obtain four subimages. The second stage of the method concerns the application of the convolutional neural network to these four images. The approach presented in this paper yields better classification performance in comparison to the results obtained by the convolutional neural network alone.

1 Introduction

Detecting and locating human faces in an image or a sequence of images are important tasks in applications like intelligent human-computer interaction, model-based coding of video sequences at very low bitrates and content-based video indexing. Given an image of arbitrary size, the task is to detect the presence of any human face appearing in the image. Face detection in complex scenes is a challenging task since human faces may appear in different scales, orientations and with different head poses. Due to change of lighting condition, facial expressions, shadows, etc., the human face appearance could change considerably. Presence of glasses is another source of variations we have to take into account.

Several approaches, such as support vector machines [10], Bayesian classifiers [10], neural networks [7][5] have been proposed so far for detection of facial regions. The face knowledge-based detector [7] first preprocess the image subwindow and applies a neural network to detect whether it contains a face. The neural network has three types of hidden units: four looking at 10x10 pixel subregions, six looking at overlapping 20x5 pixel subregions and sixteen looking at 5x5 subregions. These subregions have been chosen to represent facial features that are important to face detection in pixel windows of size 20x20. To reduce the number of false positives multiple networks are applied. They have been trained in a similar manner but under different initial conditions and with different self-selected non-face examples. A skin-color detector has been used at the preprocessing stage to limit the amount of searching. Another neural network-based approach for finding frontal faces has been presented in work [8]. The algorithm uses a modified k-means clustering algorithm to extract the six

face and non-face pattern centroids and their cluster covariance matrices from normalized input patterns. A multi-layer perceptron has been applied to classify the face and non-face patterns using different feature vectors of 12 distance measurements. The network contains 12 pairs of input units, one output unit and 24 hidden units. This work reported 79.9% to 96.3% detection rates with different data sets. The work [1] indicated that in the face recognition the face representation which is obtained on the basis of 2D Gabor filters is more robust against illumination variations than that of intensity-based. Gabor filter-based features have been used in several approaches to face recognition [10] and little work has been done to apply they to face detection. The work [5][2] presents results which were obtained during experiments with detection of faces in static images using convolutional neural networks.

Convolutional neural networks use the local receptive fields, shared weights and subsampling in order to extract and then to combine local features in a distortion-invariant manner. The feature extractor is created by the learning process and it is integrated into the classifier. The number of free parameters in a convolutional neural network is much less than in a fully-connected neural network with comparable classification capabilities due to the weight sharing. Our method uses Gabor filter-based features instead of the raw gray values as the input for a convolutional neural network to take advantage of both methods. The choice of Gabor filter responses is biologically motivated since they model the response of human visual cortical cells [3]. Gabor filters remove most of variation in lighting and contrast and can reduce intrapersonal variation. They are also robust against small shifts and small object deformations.

The remainder of the paper is organized as follows. In the next section we discuss the Gabor filter. In section 3. the components and details of convolutional neural network are presented. Section 4. reports results which were obtained in experiments. Finally, some conclusions follow in the last section.

2 Facial features extraction using Gabor filters

The main advantage of Gabor wavelets is that they allow analysis of signals at different scales, or resolution, and further they accommodate frequency and position simultaneously. The Gabor wavelet is essentially a sinewave modulated by a Gaussian envelope. The 2-D Gabor filter kernel is defined as follows:

$$f(x, y, \theta_k, \lambda) = exp\left[-\frac{1}{2}\left\{\frac{R_1^2}{\sigma_x^2} + \frac{R_2^2}{\sigma_y^2}\right\}\right] exp\left\{i\frac{2\pi R_1}{\lambda}\right\} \tag{1}$$

where $R_1 = xcos\theta_k + ysin\theta_k$ and $R_2 = -xsin\theta_k + ycos\theta_k$, σ_x and σ_y are the standard deviations of the Gaussian envelope along the x and y dimensions, λ and θ_k are the wavelength and orientation of the sinusoidal plane wave, respectively. The spread of the Gaussian envelope is defined in terms of the wavelength λ. θ_k is defined by $\theta_k = \frac{\pi(k-1)}{n}$, $k = 1, 2, ..., n$, where n denotes the number of orientations that are taken into account. For example, when $n = 2$, two values of orientation θ_k are used: 0^o and 90^o.

A Gabor filter response is achieved by convolving a filter kernel given by (1) with the image. The response of the filter for sampling point (x, y) is given by:

$$g(x, y, \theta_k, \lambda) = \sum_{u=-(N-x)}^{N-x-1} \sum_{v=-(N-y)}^{N-y-1} I(x+u, y+v) f(u, v, \theta_k, \lambda) \qquad (2)$$

where $I(x, y)$ denotes a NxN grayscale image.

In this work two different orientations and two different wavelengths are utilized. Therefore, different facial features are selected, depending on the response of each filter. In frontal or near frontal face image the eyes and mouth are oriented horizontally, while the nose constitutes vertical orientation. Fig. 1 depicts some Gabor filtered images of face samples of size 20x20. We can observe that the orientation properties of the face pattern have been highlighted. In particular, the eyes, nose and mouth have come out quite well. This does demonstrate the Gabor wavelet's capability to select localized variation in image intensity.

Fig. 1. Gabor filtered images

3 Convolutional neural architecture

A convolutional neural network [6] is a special kind of a feedforward neural network. It incorporates prior knowledge about the input signal and its distortions into its architecture. Convolutional neural networks are specifically designed to cope with the variability of 2D shapes to be recognized. They combine local feature fields and shared weights as well as utilize spatial subsampling to ensure some level of shift, scale and deformation invariance. Using the local receptive fields the neurons can extract simple visual features such as corners, end-points. These elementary features are then linked by the succeeding layers to detect more complicated features.

A typical convolutional network contains a set of layers each of which consists of one or more planes. Each unit in the plane is connected to a local neighborhood in the previous layer. The unit can be seen as a local feature detector whose activation characteristic is determined in the learning stage. The outputs of such a set of units constitute a feature map. Units in a feature map are constrained to perform the same operation on different parts of the input image or previous feature maps, extracting different features from the same image. A feature map can be obtained in a sequential manner through scanning the input image by a single unit with weights forming a local receptive field and storing the outputs of this unit in corresponding locations in the feature map. This operation is equivalent to a convolution with a small kernel. The feature map can

be treated as a plane of units that share weights. The subsampling layers which usually follow layers with local, convolutional feature maps introduce a certain level of invariance to distortions and translations. Features of decreasing spatial resolution and of increasing complexity as well as globality are detected by the units in the successive layers.

The convolutional neural network we use consists of 6 layers. Layer C1 performs a convolution on the Gabor filtered images using an adaptive mask. The weights in the convolution mask are shared by all the neurons of the same feature map. The receptive fields of neighboring units overlap. The size of the scanning windows was chosen to be 20x20 pixels. The size of the mask is 5x5 and the size of the feature map of this layer is 16x16. The layer has 104 trainable parameters. Layer S2 is the averaging/subsampling layer. It consists of of 4 planes of size 16 by 16. Each unit in one of these planes receives four inputs from the corresponding plane in C1. Receptive fields do not overlap and all the weights are equal within a single unit. Therefore, this layer performs a local averaging and 2 to 1 subsampling. The number of trainable parameters utilized in this layer is 8. Once a feature has been extracted through the first two layers its accurate location in the image is less substantial and spatial relations with other features are more relevant. Therefore layers S1 and C2 are partially connected, and the task of such a configuration is to discover the relationships between different features.

Layer C2 is composed of 14 feature maps. Each unit contains one or two receptive fields of size 3x3 which operate at identical positions within each S1 maps. The first eight feature maps use single receptive fields. They form two independent groups of units responsible for distinguishing between face and non-face patterns. The remaining six feature maps take inputs from every contiguous subsets of two feature maps in S1. This layer has 140 free parameters. Layer S2 plays the same role as the layer S1. It is constructed of 14 feature maps and has 28 free parameters. In the next layer each of 14 units is connected only to the corresponding feature map of the S2 layer. It has 140 free parameters. Finally, the output layer has one node that is fully connected to the all the nodes from the previous layer. The network contains many connections but relatively few free trained parameters. Weight sharing allows to considerably reduce the number of free parameters and improves the generalization capability.

Training of our network has been realized in a supervised manner by using the back-propagation algorithm which has been adapted for convolutional neural networks. The partial derivatives of the activation function with respect to each connection have been computed, as if the network were a typical multi-layer one. Then the partial derivatives of all the connections that share the same parameter have been added to construct the derivative with respect to that parameter.

The recognition performance of a learned system is dependent on the size and quality of the training set. The face detector was trained on 3000 non-face patches collected from about 1500 images and 1500 faces covering out-of-plane rotation in the range $-20^o, ..., 20^o$. All faces were manually aligned by eyes position. For each face example the synthesized faces were generated by random in-plane rotation in the range $-10^o, ..., 10^o$, random scaling about $\pm 10\%$, random shifting up to ± 1 pixel and mirroring. All faces were then cropped and re-scaled to windows of

size 20x20 pixels while preserving their aspect ratio, see Fig. 2. Such a window size is considered in the literature as the minimal resolution that can be used without loosing critical information from the face pattern. The training collection contains also images acquired from our video cameras. The most of the training images which were obtained from WWW are of very good quality. The images obtained from cameras are of second quality, see also the last subimage from sequence demonstrated in Fig. 2. To provide more false examples we utilized a training with bootstrapping [7]. By using bootstrapping we iteratively gathered examples which were close to the boundaries of face and non-face clusters in the early stages of training. The activation function in the network was a hyperbolic tangent. Training the face detector took around 60 hours on a 2.4 GHz Pentium IV-based PC. There was no overlap between the training and test images.

Fig. 2. Some examples from the training gallery

4 Experimental results

The experiments described in this section were carried out with a commercial binocular Megapixel Stereo Head. The depth map covering a face region is usually dense because human face is rich in details and texture. Thanks to such a property the stereovision provides a separate source of information and allows us to avoid expensive scaling down the subimages during searching for faces at all scales. A skin color detector is the first classifier in our system [4]. This fast classifier discards most of non-skin regions and therefore provides a focus of attention strategy guiding the searching for faces to only promising regions. It could detect almost all the promising regions. To find the faces the detector moves a scanning subwindow by a pre-determined number of pixels within only skin-like regions. The output of the face detector is then utilized to initialize our face/head tracker [4]. Fig. 3 depicts a typical scenario in which the face detector has been tested. The detector operates on images of size 320x240 and can process 2-5 images per second depending on the image structure.

To estimate the recognition performance we utilized only the static gray images. We obtained a detection rate of 87.5% on a test data-set containing 1000 face samples and 10000 non-face samples. Using only the convolutional network we obtaied a detection rate of 79%. This is a result of relatively simple structure of the network. It is worth to notice that such a relatively simple architecture of the network has been chosen to provide face detection in real-time using the available computational resources. The system achieves a much better recognition performance that using the convolutional neural network alone. It is much easier to train a convolutional neural network using a Gabor filtered input images than a network which uses raw images or histogram equalized images.

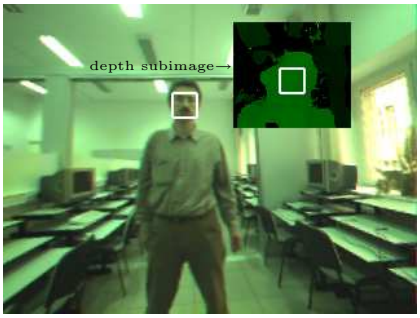

Fig. 3. Face detection

5 Conclusion

The experimental results we have obtained are very promising both in detection rates and processing speed. The Gabor filter has been used to capture efficient features for a convolutional neural network. The system achieves a much better recognition performance that using the convolutional neural network alone. The advantage of the proposed approach is that it achieves high face detection rates and real-time performance due to no exhaustive searching on the whole image.

References

1. Adini, Y., Moses, Y., Ullman, S.: Face recognition: The problem of compensating for changes in illumination direction, IEEE Trans. on Patt. Anal. Mach. Intell., vol. 19, no. 7 (1997) 721–731
2. Garcia, Ch., Delakis, M.: A neural architecture for fast and robust face detection, Int. Conf. on Pattern Recognition (2002) 44–47
3. Jones, J., Palmer, L.: An evaluation of the two dimensional Gabor filter model of simple receptive fields in cat striate cortex, Journal of Neurophysiology, vol. 58 (1987) 1233–1258
4. Kwolek, B.: Stereovision–based head tracking using color and ellipse fitting in a particle filter, 8th European Conf. on Computer Vision, LNCS 3024 (2004) 192–204
5. Lawrence S., Giles C.L., Tsoi A., Back, A.: Face recognition: A convolutional neural network approach, IEEE Trans. on Neural Networks, vol. 8, no. 1 (1997) 98–113
6. LeCun, Y., Bengio, Y.: Convolutional networks for images, speech, and time-series, In: M.A. Arbib, ed., The handbook of brain theory and neural networks, MIT Press (1995)
7. Rowley, H.A, Baluja, S., Kanade, T.: Neural network-based face detection, IEEE Trans. on Patt. Anal. Mach. Intell., vol. 20, no. 1 (1998) 23-38
8. Sung, K.K., Poggio, T.: Example based learning for view-based human face detection, IEEE Trans. on Patt. Anal. Mach. Intell., vol. 20, no. 1 (1998) 39–50
9. Yang, M-H., Kriegman, D., Ahuja, N.: Detecting faces in images: A survey, IEEE Trans. on Patt. Anal. Mach. Intell., vol. 24, no. 1 (2002) 34–58
10. Zhang, J., Yan, Y., Lades, M.: Face recognition: Eigenface, elastic matching, and neural nets, Proc. of IEEE, vol. 85 (1997) 423-435