Facial expression recognition has been widely used in human computer interaction (HCI) systems. Over the years, researchers have proposed different feature descriptors, implemented different classification methods, and carried out a number of experiments on various datasets for automatic facial expression recognition. However, most of them used 2D static images or 2D video sequences for the recognition task. The main limitations of 2D-based analysis are problems associated with variations in pose and illumination, which reduce the recognition accuracy. Therefore, an alternative way is to incorporate depth information acquired by 3D sensor, because it is invariant in both pose and illumination. In this paper, we present a two-stream convolutional neural network (CNN)-based facial expression recognition system and test it on our own RGB-D facial expression dataset collected by Microsoft Kinect for XBOX in unspontaneous scenarios since Kinect is an inexpensive and portable device to capture both RGB and depth information. Our fully annotated dataset includes seven expressions (i.e., neural, sadness, disgust, fear, happiness, anger, and surprise) for 15 subjects (9 males and 6 females) aged from 20 to 25. The two individual CNNs are identical in architecture but do not share parameters. To combine the detection results produced by these two CNNs, we propose the late fusion approach. The experimental results demonstrate that the proposed two-stream network using RGB-D images is superior to that using only RGB images or depth images.

Keywords: Facial Expression Recognition; Convolutional Neural Network; Kinect; RGB-D Images; Depth Information

1. Introduction

Human computer interaction (HCI) enables humanoid robots to assist humans in their daily lives, where the capability for sufficient interaction with human operators is a key factor [23], [24]. In daily communications, we usually use nonverbal cues, including hand gestures, facial expressions, and the tone of voice to express our feelings and opinions. A study [1] showed that facial expression constitutes 55% of the effect of a communication message. Hence, the broad vocabulary of facial expression makes it an ideal candidate for HCI. For example, in smartly controlled healthcare [2], [3], HCI systems could be considerably improved if robots could understand peoples’ emotions by analyzing facial expressions and essentially react according to their current emotional states [4].

The spatio-temporal variation of facial muscles results in the creation of various facial expressions that convey emotions [5]. Ekman and Friesen [6] defined six basic facial expression forms, i.e., anger, happiness, disgust, sadness, surprise and fear, which have been widely accepted. As early as in 1970s, Suwa et al. [7] presented a preliminary investigation on automatic facial expression analysis. They selected a series of image sequences and manually defined a fixed set of 20 points on the face. Facial
expressions were recognized by comparing these 20 points with prototype patterns. Ekman [8] developed the Facial Action Coding System (FACS) to describe the variations in facial expression by dividing faces into a number of Action Units (AUs).

Facial expression recognition is a challenging task that contributes to a wide range of applications in computer vision research. However, accurate facial expression recognition is still considered a challenge due to some environmental factors such as the illumination variations over time, cluttered background and so on.

Before expression recognition, face detection and feature extraction should be conducted. For face detection, Harr-like features [9] and joint cascade detection and alignment (JDA) [10] have been widely applied, typically including the following steps: 1) selecting an image region as a viewing window; 2) extracting some features to describe the region; and 3) determining whether the window is exactly framed a face according to feature description. For feature extraction, there are two kinds of methods namely the geometric feature-based methods and the appearance-based methods. Geometry feature-based methods are mainly based on the shape feature, which extracts facial features based on the geometric relationships between facial feature points. Recently, the methods have been utilized as Active Appearance Models (AAMs) [22]. Whereas, the appearance-based methods reflect the underlying information in images and increase discriminative characteristics in analyzing the expressions. Routray [11] classified facial expressions into six basic emotions. First, the active patches were extracted, e.g. the eyes or lips, from the facial region. Then, local binary patterns (LBP) as the feature descriptor was fed into a support vector machine for classification. For facial expression recognition, traditional approaches such as Bayesian classifiers and Support Vector Machine (SVM) perform well on classifying posed facial expressions in a controlled environment and lab settings. Liew and Yairi [17] evaluated five most commonly used feature spaces with seven classification methods in order to identify the most effective features for facial expression recognition. In [18], Weber Local Descriptor (WLD) and Histograms of Oriented Gradients (HOG) were used to capture local features in the sub-regions of an image and chi-square distance measure was used to identify facial expressions. However, these methods cannot work effectively in a spontaneous uncontrolled circumstance.

Nowadays, convolutional neural network (CNN) proposed by LeCun et al. [12] combines feature extraction and classification together. Raw data are sent into the network, and then the final result is obtained without any auxiliary process, which saves the computation consumption and meanwhile improves the recognition robustness.

Most of the existing approaches used RGB images to recognize basic facial expressions. However, RGB images acquired by conventional 2D cameras lose some useful information in the presence of unexpected illumination, occlusion and pose variations. On the contrary, RGB-D images captured by 3D sensors are invariant to pose and illumination variations. Some approaches applied 3D scanning [13] and thermal imaging [14] to recording the facial information. Though 3D scanning is accurate and invariant to illumination changes compared to other approaches, it requires specialized expensive equipment and capture in controlled environments [15]. Therefore, Microsoft Kinect as a 3D sensor is an attractive alternative due to its low cost, portability, and applicability in many interactive applications such as games and action recognition. It has been becoming one of the most widely used devices for image capture by providing both RGB information and depth information, where color information includes more details of facial expressions while depth information is insensitive to illumination conditions. Thus, images acquired by Kinect are able to preserve the diversity of facial expressions and at the same time endure the effect of illumination variations. However, only a few
researches utilized the Kinect sensor to collect image data for facial expression recognition, which results in the lack of Kinect-based RGB-D facial expression datasets.

In [15], depth information was used to recognize facial expressions with open mouth, occlusion of mouth by hand and occlusion by paper. The Gradient direction information of depth data was used as facial features and sent into the convolutional neural network for classification. Uddin and Sarkar [20] proposed an approach to recognize facial expressions from time-sequential depth videos. Local Directional Pattern (LDP) was extracted as the facial features and followed by Linear Discriminant Analysis (LDA) to make the features more robust. Finally, the features were applied with Hidden Markov Models (HMMs) for facial expression recognition. The proposed approach showed superior recognition rate against the conventional approaches.

In this paper, we proposed a novel convolutional neural network for robust facial expression recognition based on RGB-D images. Moreover, we present an RGB-D facial expression dataset acquired by Microsoft Kinect for XBOX. First, we use a Kinect sensor to capture both RGB and depth image sequences for each subject. Then, we preprocess the acquired data to make full use of the RGB and depth information. Inspired by [16], the processed data are sent into a two-stream AlexNet for classification. Beyond that, we employ data augmentation and batch normalization to prevent overfitting. Fig. 1 describes the overall schema of our architecture.

The main contributions of this paper are as follows:
(i) We introduce a novel convolutional neural network (CNN) architecture to automatically recognize facial expressions. The architecture is composed of two separate CNN processing streams: one is for RGB images and the other is for depth images, which are consecutively combined with a late fusion network. Face images are resized to 227*227 pixels because taking large-size images as input will result in a deeper network, which can extract more meaningful features from the images;
(ii) We collect a new RGB-D facial expression dataset. For each subject, image sequences of unspontaneous expression of six basic emotions (along with the neutral expression) are acquired;
(iii) We preprocess depth images by pixel filtering and rendering depth data into grayscale;
(iv) We employ data augmentation and batch normalization to avoid overfitting. For data augmentation, we first randomly crop some images from the original dataset, and then randomly rotate, flip, and shift the images to expand the number of images in the datasets. Batch normalization can reduce intra-class variation while does not distort the images.

The rest of this paper is arranged as follows. Section 2 describes the newly proposed network. Section 3 describes the data acquisition procedure. Section 4 presents the experimental results and Section 5 concludes.

2. A Two-Stream Architecture for RGB-D Facial Expression Recognition

2.1. Data preprocessing
Since the RGB and depth data were usually collected under a messy background and the raw depth data are very noisy. That is, many pixels in the images may have no depth values due to multiple reflections, transparent objects or scattering in certain surfaces, and thus the acquired RGB and depth data cannot be directly fed into the designed deep neural network. In order to make the network perform better, raw data need to be processed before use. In this paper, we preprocess image data by depth-pixel filtering, grayscale transformation, face detection, and data augmentation.

2.1.1. Preprocessing of depth images

Objects closer to the Kinect sensor are lighter in color and those further away are darker. The highest resolution for depth data is 512*424, but even at this resolution, the image quality looks poor indeed because of image noisy. The noise manifests itself as white spots continuously popping in and out of the image. Some of the noise in the data comes from the IR light being scattered by hitting an object; some comes from the shadows of the objects closer to the Kinect. This problem can degrade useful image information and affect the network performance. To this end, we use pixel filtering to smooth the depth data.

For depth images, we look for zero values that cannot be processed properly by the Kinect. Then, we remove as many of them as realistically possible without degrading other meaningful features of the data. Firstly, we consider a zero value as a candidate for filtering and determine whether it meets the filtering criterion defined by the neighboring pixels. The filter has two “bands” around the candidate pixel and uses it to search for non-zero values in other pixels. As seen in Fig. 2a, the filter creates a frequency distribution of these values and compute how many pixels are found in each band. After that, it compares these non-zero values with the threshold for an arbitrary band to determine if the candidate should be filtered. If the number of non-zero values in any band exceeds the threshold, the statistical mode of all the non-zero values in both bands is applied to the candidate; otherwise no operation is needed. Fig. 2b is an example of pixel filtering. The average number of non-zero pixels in the region is 1583 and the statistical mode is 2000. Suppose the threshold is 6, we apply the statistical mode 2000 to the candidate pixel since the number of non-zero values exceeds the threshold.

In order to facilitate the observation, we use blue color to denote the pixels with zero values. Fig. 3a shows the original image and Fig. 3b gives the filtered image. As we can see, the number of pixels with zero values has decreased a lot.
Fig. 3. a shows the original image and b gives the filtered image.

The depth data essentially encode the distance of objects from the Kinect sensor. They cannot be directly used as the input to the convolutional neural network (CNN). To solve this problem, we render the depth data into grayscale and replicate the grayscale values to three channels (i.e., the first two channels represent the image pixel and the third channel shows the color information) required by the network input. Fig. 4 shows depth images after pixel filtering and rendering to grayscale. After preprocessing, they can be input to the CNN.

Fig. 4. From left to right, there are the original depth image, the depth image after pixel filtering, and the depth image after rendering to grayscale, respectively.

2.1.2. Preprocessing of RGB images

Image normalization is widely used in object detection, medical image analysis, person re-identification, etc. The principle of normalization is to convert the original image into a standard mode through a series of changes, e.g., translation, rotation, scaling, etc. Due to the influence of light intensity or shadow in different time and circumstances, the RGB images we collected are different in modes, which may result in the loss of important information. Therefore, in image processing, we usually normalize RGB images in order to eliminate the influence of light and shadow. The normalization of RGB images is given as follows:

\[ r = \frac{R}{R + G + B} \]  
\[ g = \frac{B}{R + G + B} \]  
\[ b = 1 - r - g \]

Here, R, G, B represents the pixel value of a point and r, g, b represents the pixel value of the point
after normalization, respectively. For example, due to illumination changes, the pixel value of point A changes from RGB (30, 90, 120) to RGB (45, 135, 180) at different time. According to Eq. (1)-(3), the pixel values of point A at different time are rgb (1/8, 3/8, 1/2) and rgb (1/8, 3/8, 1/2) respectively, which remains the same. That is, through image normalization, the effects of light intensity or shadow can be eliminated.

2.1.3. **Face detection**

We annotate our collected dataset by “neutral”, “anger”, “disgust”, “fear”, “happy”, “sad”, and “surprise”. As shown in Fig. 5, the original image was captured in complicated background. As a result, directly using the deep neural network not only extracts facial features but also extracts a lot of unnecessary information that reduces the recognition accuracy. Thus, accurate face detection is highly correlated with the recognition accuracy. In order to detect the face, OpenCV face recognition is used to read images from the dataset, and then detect the face and frame the face with red border. After face detection, the influence of messy background is removed while useful information is preserved to the maximum extent.

![Face detection](image)

Fig. 5. Face detection for the happiness expression. The top row shows the RGB image and the detected face image; the bottom row shows the depth image and the detected face.

2.1.4. **Data augmentation**

It is important to consider overfitting in deep convolutional neural networks since overfitting makes the model perform too well on the training data while leading to poor performance in the validation data and the testing data. In this paper, we use three methods to augment the original data. Firstly, we randomly crop images with the size 227*n or n*227 from our image dataset. This yields a large number of additional training samples. Secondly, we flip the images horizontally to capture the reflection invariance. Finally, we randomly shift the images to capture the translation invariance.

2.2. **A two-stream convolutional neural network**

In this sub-section, we present a two-stream convolutional neural network (CNN) based on AlexNet [16] for facial expression recognition from RGB-D data. The advantages of AlexNet are: 1) AlexNet can extract image features more effectively with less computation resources and running time; 2) AlexNet is the first network that proposed dropout to avoid overfitting; 3) it uses maximum pooling to avoid the blurring effect of average pooling; and 4) it selects ReLU as the activation function to solve
the gradient dispersion problems. The network architecture is illustrated in Fig. 6, which consists of two CNN streams operating on color images and depth images, respectively. The two sub-networks are then automatically combined in a late fusion approach. We feed the preprocessed RGB/depth images with the size of 227*227 as the input, which is then followed by a series of layers including five convolution layers, three max pooling layers, two fully-connected layers, a fusion layer and a soft-max layer. We add a batch normalization layer to each of the convolution layers to avoid overfitting and add ReLU activation function to all of the batch normalization layers to avoid the vanishing gradient problem.

Fig. 6. The architecture of the proposed two-stream CNN.

3. Data Acquisition

In this section, we use Microsoft Kinect to capture both RGB and depth data. In this section, we first introduce the Kinect sensor, and then give the details of our data acquisition process. In summary, the dataset contains 210 RGB-D image sequences for seven expressions collected from 15 (9 males and 7 females) graduate students in indoor environment with different illuminations.

3.1. Kinect sensor

The Kinect sensor was primarily designed for natural interaction in computer game environment, which incorporates several advanced sensing hardware, including a general color/RGB camera, a depth sensor (an infrared camera and an infrared projector), and a four-microphone array, as shown in Fig. 7. In this way, it can provide RGB images, depth images, and audio signals simultaneously. Several useful software tools allow users to develop products for various applications, such as 3D human motion, hand gesture recognition, person re-identification, facial expression recognition, etc.
In this paper, Kinect refers to both RGB/depth sensing hardware and the software-based technology that interprets the RGB/depth signals. The RGB camera delivers three basic color components of the video. It can produce images running at 30 frames/s at the highest resolution of 1280*1024 pixels. The depth sensor consists of an IR camera and an IR laser projector. The IR laser projector casts an IR speckle dot pattern into the 3D scene while the IR camera captures the reflected IR speckle [21]. That is, Kinect is an instance of a structured lighting depth sensor. It has a practical ranging limit of 0.5m-4.5m distance and can output videos at a frame rate of 30 frames/s with the resolution of 512*424 depth pixels. Fig. 8 shows an RGB image and its corresponding depth image captured by the Kinect sensor.

3.2. Data acquisition

In the past decade, automatic facial expression recognition has been extensively studied using 2D static RGB images and 2D video sequences. However, RGB images are sensitive to illumination conditions and pose changes, which affects the recognition results. In contrast, depth information is inherently tolerant to illumination variations and pose, and thus using depth information to improve the performance of facial expression recognition becomes a trend.
We collected an RGB-D facial expression dataset using the Kinect for Xbox sensor in unspontaneous scenarios. The sizes of the original color and depth images are 1280*1024 and 512*424, respectively. We annotated the dataset with seven expressions (i.e., neutral, anger, disgust, fear, happy, sad, and surprise) for 15 subjects (9 males and 6 females) aged from 20 to 25. All of the participants are graduate students from Nanchang University, China. For each subject, we captured his/her facial expressions twice, i.e., there are 15*2=30 sets of image sequences for each expression, and there are 210 image sequences for seven expressions in total. To record unspontaneous facial expressions, we asked the participants to seat about 60cm in front of the Kinect sensor and make sure their faces are in the middle of the computer screen. The environment of RGB-D data acquisition is given in Fig. 9. Afterwards, we asked the subjects to show certain expressions by giving them some instructions, which includes verbal instructions, images, and live demonstration. We recorded color and depth image sequences simultaneously, each of which starts from a neutral expression, gradually reaches the peak of expression, and finally returns to the neutral expression, as seen in Fig. 10.

We select the most representative image frames for each expression depending on its amplitude. After image preprocessing and data augmentation, we obtained 9800 RGB-D images and then divide them into the training set and the testing set. Among them, 8400 images are used for training and the remaining 1400 images are used for testing. Detailed information can be found in Table 1.
<table>
<thead>
<tr>
<th>Datasets</th>
<th>Expressions</th>
<th>Neutral</th>
<th>Anger</th>
<th>Disgust</th>
<th>Fear</th>
<th>Happy</th>
<th>Sad</th>
<th>Surprise</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training set (RGB images)</td>
<td></td>
<td>1200</td>
<td>1200</td>
<td>1200</td>
<td>1200</td>
<td>1200</td>
<td>1200</td>
<td>1200</td>
</tr>
<tr>
<td>Training set (Depth images)</td>
<td></td>
<td>1200</td>
<td>1200</td>
<td>1200</td>
<td>1200</td>
<td>1200</td>
<td>1200</td>
<td>1200</td>
</tr>
<tr>
<td>Testing set (RGB-D images)</td>
<td></td>
<td>200</td>
<td>200</td>
<td>200</td>
<td>200</td>
<td>200</td>
<td>200</td>
<td>200</td>
</tr>
</tbody>
</table>

4. Experimental Results

We evaluate the newly proposed two-stream convolutional neural network on our self-collected dataset as mentioned in Section 3. After image preprocessing and face detection, we resize the face images to 227*227 pixels because large-size images as input will result in a deeper network which can extract more meaningful features from the images. The model is trained on one GTX1080 TI GPU with an initialized learning rate 0.001 and the batch size is around 128~256. We add batch normalization after each convolutional layer. The model obtains the optimal classification accuracy of 99.5% after 300 epochs. We conduct three sets of experiments for comparison. Experiment 1 is to evaluate the effect of using depth images; Experiment 2 is to compare our facial expression recognition method with some representative CNN-based methods; Experiment 3 is the comparative study between the raw data and the pre-processed data.

**Experiment 1:** In order to demonstrate that using both RGB images and depth information can improve the expression recognition rate, we provide the recognition results as follows. Fig. 11 shows the training and testing accuracy curves on three sub-datasets, where (a) is for RGB-D images, (b) is for RGB images, and (c) is for depth images. As we can see, the network on the RGB-D dataset achieves the best recognition accuracy of 99.5%. For the RGB dataset, the recognition accuracy is 95.92%, which is 3.58% lower than using the RGB-D dataset. For the depth dataset, the recognition rate is only 58.64%. This is because there are some challenges involved in depth information generated from the Kinect depth sensor which may result in the low accuracy for facial expression recognition. The first challenge is that the depth images captured by the Kinect depth sensor are relatively of low-quality. The second challenge is from the ambiguous edges of the objects.

![Fig. 11. The training and testing accuracy curves on RGB-D images, RGB images, and depth images, respectively.](image1)

**Experiment 2:** We use AlexNet to conduct facial expression recognition. In order to demonstrate its superiority to other representative CNN-based methods, we carry out the following comparison experiment. As seen in Fig. 12, (a) is the architecture based on AlexNet and (b) is the architecture based
on VGGNet [19]. The recognition accuracy of these two networks is 99.5% and 95.7% when using RGB-D images. Fig. 12 also shows that the AlexNet-based network is more robust. Meanwhile, both the number of network parameters and the training time of AlexNet are less than using VGGNet.

![Fig. 12. The training and testing accuracy curves using AlexNet and VGGNet, respectively.](image)

**Experiment 3:** We conduct the comparative study between the original RGB-D data and the RGB-D data after data preprocessing. As shown in Fig. 13, the system obtains the accuracy on the preprocessed RGB-D data with 99.5% and only 88.9% on the original un-preprocessed RGB-D data. It indicates that the network with data preprocessing can get 10.6% performance gains. The low accuracy on the raw RGB-D dataset is mainly due to the effect of messy background and noise.

![Fig. 13. The training and testing accuracy curves on RGB-D data before and after data preprocessing.](image)

Fig. 14 shows the confusion matrices of our method on different datasets. As seen in Fig. 14(a), very high recognition rates are achieved on the RGB-D dataset, where the recognition rates of “angry”, “Disgust”, “Happy” and “Surprise” are 100%, respectively. While the lowest rates occur on “Fear” with the recognition rate of 98.5%. This is mainly because different subjects express “Fear” in quite a different way, and some expressions are not so obvious. In Fig. 14(b), a recognition rate of 99% is achieved for “neutral” and “happy” on the RGB dataset. In Fig. 14(c), the highest and the lowest recognition rates on the depth dataset are 99.5% and 32.5%, which belong to “anger” and “surprise”, respectively.

![Fig. 14. The confusion matrices of our method on different datasets.](image)
5. Conclusions and Future Work

This paper presents a two-stream convolutional neural network (CNN) for facial expression recognition based on our self-collected and annotated RGB-D dataset using the Kinect for XBOX in unsupervised scenarios. In previous studies, most of the systems used RGB images for facial expression recognition, but the main limitations are the problems associated with variations in pose and illumination. In order to solve the problems and improve the recognition accuracy, we build a new RGB-D facial expression dataset and apply several methods to preprocessing the collected images. For the RGB images, we normalize the original images to a standard mode; for the depth images, we filter them and render them into grayscale. After that, we detect the face from both RGB and depth images, and finally augment the number of images to avoid overfitting. In order to evaluate our proposed facial expression recognition method, we present the two-stream CNN architecture separately operating on color and depth information. The two sub-networks are then automatically combined in a late fusion approach. The experimental results demonstrate that the network with RGB-D images is superior to the one with only RGB images or depth images.

However, the size of our dataset is still small and the network is only applied to 2D static images. In the future, we will increase the size of our dataset by including more individuals of different ages. Moreover, we will improve the network for real-time facial expression recognition. Last but not the least, we will consider extending the approach in this article to make it more applicable to human-computer interaction systems.

6. Acknowledgements

This work is supported by National Natural Science Foundation of China under Grant 61703198 and 51575412, Natural Science Foundation for Distinguished Young Scholars of Jiangxi Province under Grant 2018ACB21014, Open Fund of Key Laboratory of Metallurgical Equipment and Control Technology of Ministry of Education, Wuhan University of Science and Technology under Grant 2017B02, National Key R&D Program of China (Grant No. 2018YFB1304600), CAS Interdisciplinary Innovation Team (Grant No. JCTD-2018-11).

7. Reference


(CVPR, 2010), pp. 511-518.
