TSINGHUA SCIENCE AND TECHNOLOGY
ISSN 1007-0214 04/12 pp275–286
DOI: 10.26599/TST.2019.9010035
Volume 26, Number 3, June 2021

Deep Learning-Based Classification of the Polar Emotions of "Moe"-Style Cartoon Pictures

Qinchen Cao, Weilin Zhang, and Yonghua Zhu*

Abstract: The cartoon animation industry has developed into a huge industrial chain with a large potential market involving games, digital entertainment, and other industries. However, due to the coarse-grained classification of cartoon materials, cartoon animators can hardly find relevant materials during the process of creation. The polar emotions of cartoon materials are an important reference for creators as they can help them easily obtain the pictures they need. Some methods for obtaining the emotions of cartoon pictures have been proposed, but most of these focus on expression recognition. Meanwhile, other emotion recognition methods are not ideal for use as cartoon materials. We propose a deep learning-based method to classify the polar emotions of the cartoon pictures of the "Moe" drawing style. According to the expression feature of the cartoon characters of this drawing style, we recognize the facial expressions of cartoon characters and extract the scene and facial features of the cartoon images. Then, we correct the emotions of the pictures obtained by the expression recognition according to the scene features. Finally, we can obtain the polar emotions of corresponding picture. We designed a dataset and performed verification tests on it, achieving 81.9% experimental accuracy. The experimental results prove that our method is competitive.

Key words: cartoon; emotion classification; deep learning

1 Introduction

The cartoon animation industry has developed into a huge industrial chain with a large potential market involving games, digital entertainment, and other industries. With the continuous development of the cartoon animation market, increasing amounts cartoon materials have been piled up as unprocessed, raw materials. Many cartoon animation companies collect these materials through various ways and store them as reusable references to help their animators create better cartoon animation work. These raw materials require analysis, tagging, and classification before they can actually be applied as useful materials. However, current cartoon materials are usually loosely classified based on objective attributes rather than subjective requirements. Hence, searching the database of materials to find satisfying references has become a time-consuming and laborious task. Among the required analyses, sentiment polarity classification is based on subjective experience rather than objective content, which is important for animators to decide whether a piece of cartoon picture material is suitable for his work.

Numerous algorithms have been proposed in recent years to deal with cartoon pictures. Huo et al.[1] proposed a benchmark for caricature recognition, while Klare et al.[2] proposed a Sketch recognition algorithm that is used to obtain recognition information from the sketch and local area of the digital face image. Ouyang et al.[3] solved the problem of matching comics with

• Qinchen Cao and Weilin Zhang are with the School of Computer Engineering and Science, Shanghai University, Shanghai 200444, China. E-mail: {caoiqn00123, zeroized}@shu.edu.cn.
• Yonghua Zhu is with the Shanghai Film Academy, Shanghai University, Shanghai 200072, China. E-mail: zyh@shu.edu.cn.
∗ To whom correspondence should be addressed.
Manuscript received: 2019-07-21; accepted: 2019-07-28

photos by defining a set of qualitative facial features for comics and photos. Most of these methods focus on the classification, labeling, or expression recognition of cartoon materials. In comparison, only a few studies have been published on emotion recognition for cartoon materials. As the sentiment analysis of cartoon materials helps speed up the cartoon enterprise's screening process of material to a certain extent, we designed a polar emotion classification algorithm that is suitable for part of the cartoon materials based on the unique characteristics of such materials.

Owing to the richness of Japanese cartoon materials, we choose Japanese-style cartoon pictures as our research object, that is, "Moe"-style cartoon pictures. In our polar sentiment classification algorithm, we consider two main issues: facial feature extraction and scene feature extraction. Based on the characteristics of the Moe drawing style according to the related knowledge of existing comics and cartoons, our algorithm models the face of the Moe-style cartoon character, uses five feature points to express the positions of the facial features in a simple way, and designs the algorithm model for the five senses. We can obtain the emotional nuances of cartoon faces by feature matching. Second, the scene of cartoon pictures is also used as the basis of sentiment analysis. Finally, we obtain the polar emotions of cartoon pictures based on the results of expression and scene analyses.

2 Related Work

With the development of deep learning, it has been applied in various fields of computer vision and emotion classification. Image emotion classification methods can be roughly divided into three types based on emotion detection, visual emotion ontology, and deep learning.

The method of emotion detection is the first to learn from the method of emotional semantic image retrieval, which uses the low-level features of an image to detect the emotions contained therein. Siersdorfer et al.[4] were the first to conduct image sentiment classification. The text first divides the image into 16 blocks and extracts the corresponding color histograms for 16 blocks. The color features of the full image contain certain positional information, and the differential Gaussian pyramid is used to extract the sharp points in each image. Then, we extract the Scale-Invariant Feature Transform (SIFT) features in the region, so that the color features and SIFT features together form the characteristics of the image. The text metadata carried by the image are also used in the text and SentiWordNet is used to positively and negatively score the text information. Finally, the Support Vector Machine (SVM) and naive Bayesian methods are used to classify all the above features. Li et al.[5] combined the global features of the image with the local features, using the color, texture, and SIFT features of the image as the global features of that image. For local features, the image segmentation algorithm is first used to segment the image; then, for each segment after segmentation, color features, color emotion features, and texture features are used as the local emotional features of the image. Finally, after the global feature is subjected to principal component analysis dimension reduction, the sparse coding technique is used to establish the relationship between the global and local features. Finally, the emotion classification is performed.

Due to the use of low-level emotional features, many processes are required to extract the emotional features of the image, which makes the process time-consuming and unable to satisfy the emotional classification of large-scale image data. Thus, people have attempted to use the medium features of the image to conduct the classification. Borth et al.[6] and others first proposed a groundbreaking idea using the method of visual sentiment ontology to emotionally classify images, with the aim of satisfying large-scale image data processing. Borth et al.[6] noticed a gap between the low-level image features and the emotional features reflected by the images. In order to compensate for the gap between them, a medium feature representation method based on visual sentiment ontology is proposed. First, based on the 24 emotions mentioned in Plutchik's emotional theory, some search keywords are derived to retrieve images and videos on platforms, such as Flickr and YouTube, and these strong emotions are extracted based on the images and videos retrieved. The adjectives and all the nouns form the corresponding Adjective Noun Pairs (ANPs). The ANPs are then ranked according to the frequency of occurrence of all ANPs, until we finally obtain 1200 ANPs. Finally, the corresponding test is trained with the corresponding images containing the ANPs. We use SentiBank to classify image emotions. The ANP method proposed in the article has a strong correlation with the concept of emotion and has achieved an accuracy rate beyond the simple text sentiment classification, which has a high research value.

In the process of image sentiment classification, different low-level image descriptions and medium feature attributes can have great influence on image sentiment classification; the main disadvantage of this approach is that numerous psychological and aesthetic aspects are needed in the process of training. Knowledge and manual intervention are needed to define medium characteristics and to the fine-tune the emotional classification results, respectively.

Meanwhile, for object recognition and image classification tasks, great achievements have been made despite the artificially established image description features, such as color histogram, Histogram of Oriented Gradient (HOG), SIFT, and other features. Nevertheless, the effects have been greatly improved with the development of deep learning and the application of image classification. Due to the excellent performance of deep learning, researchers have gradually began to use it to extract image features, but a large amount of data are needed to train the network because there are too many parameters within such a network. Xu et al.[7] used the ILSVRC-2012 dataset to train Convolutional Neural Networks (CNN) because the annotated sentiment classification data set is small and does not adequately train CNN. After training CNN, the model is used to extract the sentiment classification of the picture and then the Logistic Regression (LR) classifier is trained by using the annotated image sentiment data set; afterwards, the image sentiment classification can be performed. Compared with Borth's method[6], our proposed method can better perform based on medium image features by about 13%. Meanwhile, due to the lack of image sentiment data, Xu et al.[7] first used the ImageNet dataset to train the network and then transferred the parameters to fine-tune the parameters with the sentiment dataset, which also achieved good results. Campos et al.[8] used the trained AlexNet parameters to initialize the network due to the small image sentiment annotation dataset. If the network is to be used, a layer of two neurons should be added to the AlexNet network. The parameters of this layer are generated by randomization, and during the training process, the learning rate of this layer is larger than those of other layers, which makes it convenient for adjusting the parameters in place as soon as possible.

Some studies about animation have also been published. For example, de Juan and Bodenheimer[9] proposed a method for creating novel animations from a library of existing two-dimensional (2D) cartoon data. With a small amount of cartoon data, they first employed the nonlinear dimensionality reduction method to discover a lower-dimensional structure of the data. Then, the user selected a start and end frame and the system traversed this lower-dimensional manifold to resequence the data into a new animation. The system can automatically detect when a new sequence has visual discontinuities and may require additional source materials. Wang and Raj[10] reported that manifold learning is a popular animated machine learning technique that is closely related to animation research. Manifold learning is an unsupervised learning method that converts high-dimensional into low-dimensional data[10].

Some techniques for face recognition have been introduced in the literature. Bhatt et al.[11] proposed an automated algorithm to extract discriminating information from local regions of both sketches and digital face images. They first used multiscale circular Weber local descriptor to present local facial regions. Then, they employed an evolutionary memetic optimization algorithm to assign an optimal weight to every local facial region to boost the identification performance. Given that forensic sketches or digital face images can be of poor quality, a preprocessing technique is adopted to enhance the quality of images and improve the identification performance. Comprehensive experimental results show that the proposed algorithm yields better identification performance compared with the existing face recognition and two commercial face recognition systems. Klare et al.[2] addressed the problem of identifying a subject from a caricature. In their paper, they proposed a set of qualitative facial features that encodes the appearance of both caricatures and photographs. They also utilized crowdsourcing to assist in the labeling of the qualitative features. Then, they combined LR, multiple kernel earning, and support vector to generate a similarity score between a caricature and a facial photograph based on the obtained features. After conducting experiments on a dataset of 196 pairs of caricatures and photographs, their results indicate that their proposed method helped leverage the ability of humans to recognize caricatures to improve automatic face recognition methods. Ruiz-Garcia et al.[12] explored two CNN architectures that offer automatic feature extraction and representation, followed by fully connected softmax layers to classify

images into seven emotions. The first proposed architecture produces state-of-the-art results with an accuracy rate of 96.93% and the second architecture with split input produces an average accuracy rate of 86.73%, respectively.

Meanwhile, Abaci and Akgul[13] addressed the problem of matching caricatures to photographs by defining a set of qualitative face features—both for caricatures and photographs—wherein features are automatically extracted from photos and manually labeled in caricatures. The experimental results show the good performance of their proposed approach[13]. Ouyang et al.[3] took a different approach and explored learning a mid-level representation within each domain that allowed faces in each modality to be compared in a domain-invariant way. In particular, they investigated sketch-photo face matching and went beyond the well-studied viewed sketches to tackle forensic sketches and caricatures where representations are often symbolic. They approached this by learning a facial attribute model independently in each domain that represents faces in terms of semantic properties. Thus, this representation is more invariant to heterogeneity and distortions and robust to misalignment. Next, they integrated synergistically the intermediate level attribute representation with the original low-level features using Canonical Correlation Analysis (CCA). Their framework shows impressive results on cross-modal matching tasks using forensic sketches and even more challenging caricature sketches[3].

Crowley et al.[14] presented a framework called Local Feature-based Discriminant Analysis (LFDA) to identify forensic sketches. In LFDA, they individually represented both sketches and photos using SIFT feature descriptors and multiscale local binary patterns. Multiple discriminant projections are then used on the partitioned vectors of the feature-based representation for minimum distance matching. They applied this method to match a data set of 159 forensic sketches against a mug shot gallery containing 10 159 images. Compared with a leading commercial face recognition system, LFDA offered substantial improvements in matching forensic sketches to the corresponding face images. Crowley et al.[14] also further improved the matching performance using race and gender information to reduce the target gallery size. Additional experiments demonstrated that the proposed framework can ensure state-of-the-art accuracy when matching viewed sketches.

3 Methodology

In our polar sentiment classification algorithm, taking into account the nature of the cartoon picture, we split the picture sentiment analysis process into two parallel parts. On the one hand, we extract the face part of the person and analyze its expression; on the other hand, we analyze the overall scene of the picture and then derive the results of the sentiment analysis based on the data obtained from the two processes. The overall framework of the proposed model is shown in Fig. 1.

3.1 Facial part

3.1.1 Cartoon character facial extraction

We use Multi-Task Cascaded Convolutional Networks (MTCNN) to locate the faces of the cartoon characters. MTCNN, a multi-task, cascade-based face detection framework proposed by Zhang et al.[15], can perform face detection and face feature point detection simultaneously. The MTCNN first transforms the input image to different scales and constructs an image pyramid to adapt to the detection of faces of different sizes, and then the reconstructed image is processed by Proposal Network (P-Net), Refine Network (R-Net), and Output Network (O-Net) in order. P-Net has a basic structure consisting of a fully connected network. For the image pyramid completed in the first step, the preliminary feature extraction and calibration frame are performed by a Fully Convolutional Networks (FCN), after which the Bounding Box Regression adjustment window and the Non-Maximum Suppression (NMS) are used to filter most of the windows. R-Net has a basic structure consisting of a convolutional neural network. Compared with the first layer of P-Net, a fully connected layer is added, so the screening of

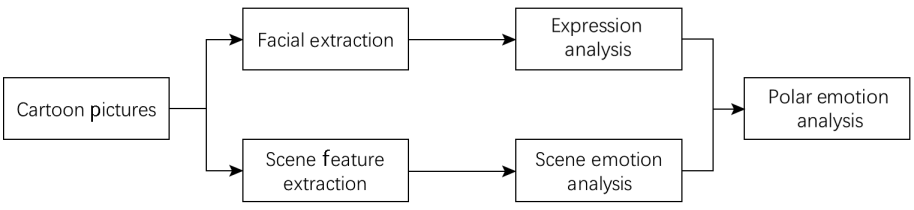

Fig. 1 Overall framework of the proposed model.

input data becomes stricter. After the picture passes P-Net, many prediction windows are left. We send all the prediction windows to R-Net. This network then filters out a large number of candidate frames with poor effect, before finally bounding the selected candidate frames. Box Regression and NMS further optimize the forecast results. Meanwhile, O-Net has a basic structure consisting of a relatively complex convolutional neural network, which has a convolutional layer compared with R-Net. The difference between O-Net and R-Net is that this layer structure can recognize the area of the face through more supervision; the former can also return the facial features of the person and finally output the five facial features in the detection of the faces with different sizes. In a word, P-Net performs the preliminary feature extraction. Then, R-Net filters out a large number of candidate frames with a weak effect. O-Net recognizes the area of the face through more supervision and then return the facial features of the person, before finally producing the output consisting of five facial features.

Through a different loss function structure design, each network implements different functions on the application side. The loss function L that needs to be minimized in training comes from three aspects as detailed below.

Face classification. For each sample x_i, we calculate the cross entropy loss function given by

$$L_i^{\text{det}} = -(\gamma_i^{\text{det}} \log(p_i) + (1 - \gamma_i^{\text{det}})(1 - \log(p_i))) \quad (1)$$

where p_i is the probability that the network predicts that x_i is the face, and $\gamma_i \in \{0, 1\}$ denotes the ground-truth label. This cross entropy loss function expresses the degree of "predicting the probability of the face" and "the fact that the face is not the face". The closer it is, the smaller the entropy is, and the smaller the loss generated. Then the target is expressed as $\min(L_i)$.

Bounding box regression. For each sample x_i, calculate the Euclidean distance:

$$L_i^{\text{box}} = \|\hat{\gamma}_i^{\text{box}} - \gamma_i^{\text{box}}\|_2^2 \quad (2)$$

where γ^{box} is a four-tuple consisting of upper left corner x, upper left corner y, height, andd width. $\hat{\gamma}^{\text{box}}$ is predicted and the second γ^{box} is real. The closer the predicted bounding box is to the real one, the smaller the Euclidean distance. The target is expressed as $\min(L_i)$.

Facial landmark localization. For each sample x_i, calculate the Euclidean distance:

$$L_i^{\text{landmark}} = \|\hat{\gamma}_i^{\text{landmark}} - \gamma_i^{\text{landmark}}\|_2^2 \quad (3)$$

where γ^{landmark} is a tuple containing 10 elements: x, y of the left eye; x, y of the right eye; x, y of the nose; x, y of the left point of the mouth; and x, y of the right point of the mouth. $\hat{\gamma}_i^{\text{landmark}}$ is predicted and $\gamma_i^{\text{landmark}}$ is real. The closer the predicted landmarks are to the real one, the smaller the Euclidean distance. The target is expressed as $\min(L_i)$.

Since the above three loss functions are not calculated for each input during training, Eq. (4) is defined.

$$\min \sum_{i=1}^{N} \sum_{j \in (\text{det, box, landmark})}^{n} \alpha_j \beta_i^j L_i^j \quad (4)$$

where N is the number of training samples and α_j indicates the importance of the task. In the article, we use $\alpha_{\text{det}} = 1, \alpha_{\text{box}} = 0.5$, and $\alpha_{\text{landmark}} = 0.5$ in P-Net and R-Net, and $\alpha_{\text{det}} = 1, \alpha_{\text{box}} = 0.5$, and $\alpha_{\text{landmark}} = 1$ in O-Net. Facial landmark positioning is more accurate; here $\beta_j \in \{0, 1\}$ is a sample type indicator.

In our experiments, the drawings of the Moe-style characters are generally based on the standard proportions of the human body. As shown in Fig. 2, the face of the Moe-style cartoon character is constructed according to the proportion of "three courts and five eyes", which is the general standard ratio of the face length and face width of a person. If it matches this ratio, even if the facial features are not perfect, the image looks very comfortable and beautiful. Otherwise, it may not be able to fully represent the ideal face. As the facial features of the character are fixed, it is easy for us to obtain the approximate position of the facial features in the case where the face position has been determined.

3.1.2 Face parsing

The facial expressions of the Moe-style cartoon pictures are constructed through six parts from top to bottom:

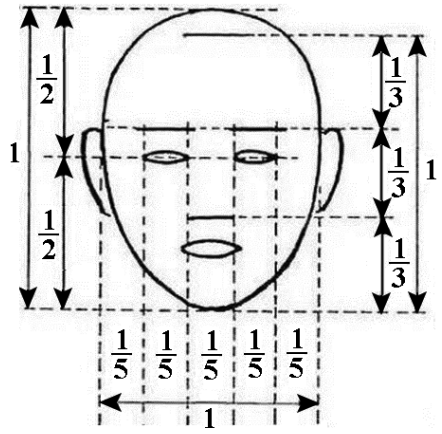

Fig. 2 Three courts and five eyes.

eyebrows, upper eyelids, pupils, lower eyelids, cheeks, and mouth. Each part has a different response to the mood change. For example, the pupils enlarge when a person is happy and shrink when a person is scared. These are more obviously shown in the cartoons after being enlarged.

In general, the primary emotions represented by the expressions are of six kinds: fear, anger, disgust, happiness, sadness, and surprise. The more complex emotions are basically a combination of these emotions, and these complex feelings are expressed through "expression addition". Of course, the "addition" mentioned here is not a general addition, and there is no positive or negative offset. For example, in the case of sad+happy, when sorrow is greater than joy, it may be sad to laugh at yourself; when happiness is greater than sadness, it can be understood as crying for joy. By analyzing the emotions and the intensity of the eyebrows, upper eyelids, pupils, lower eyelids, cheeks, and mouths, we can easily analyze the feelings expressed on the face.

This paper aims to extract the common features of facial expressions and score the emotional tendencies of each part. The scoring is a vector, that is, there may be multiple emotional scores in one facial expression. As shown in Fig. 3, we can see the slanted eyebrows in both the expressions of anger and joy. The scores here are derived from statistics. The feature is divided into three levels: strong, medium and weak, according to the frequency of appearance of the same feature in the different expressions of the dataset and then the corresponding score is given. For each class of mood, the weight of each facial part is determined by its correlation with the mood. For example, in the "happy" pictures, if the max ratio of all kinds of mouths is 40%, the initial weight before the normalization of the mouth in the "happy" class is 0.4. Then, each class is normalized, and the overall weight of a facial part is the average weight of this part in all the classes of mood. As we only discuss the polar emotions of cartoon pictures here, we reduce the emotional set to positive, neutral, and negative emotions. We map the happiness and surprise to positive emotions and other emotions to negative emotions. We also optimize the analysis process, the eyebrows are split and the eyelids are merged into the eyes. The weights of the final parts and the scores of the individual features are shown in Table 1. Given that the data in Table 1 are obtained through statistical data in the experimental data set, this model has certain limitations.

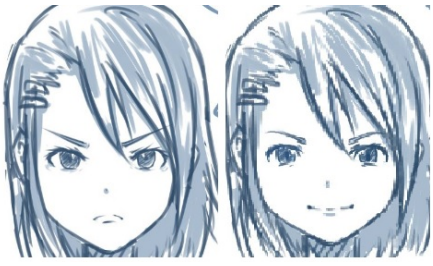

Fig. 3 Slanted eyebrows in different expressions.

In summary, the emotion expressed by the expression (E_1, E_2) is the sum of the product of the weight and the score (e_1, e_2) of each part:

$$E_i = \sum \varepsilon_{\text{part}} e_i, i \in \{1, 2\} \tag{5}$$

where $\varepsilon_{\text{part}}$ is the weight of the facial part, e_1 is the positive score, and e_2 is the negative score. As the value of the quantized value of each part is customized, the values of the final results E_1 and E_2 are weak. Here the difference ΔE of E_1 and E_2 is selected as the reference of the judgement of emotion.

$$\Delta E = E_1 - E_2 \tag{6}$$

According to the experimental result, ΔE is a neutral emotion between -0.3 and 0.3, a positive emotion greater than 0.3, and a negative emotion less than -0.3.

Table 1 Emotion weight and feature score.

Facial part	Feature	Score	Weight
Left eyebrow		(0.5, 0.2)	0.1
		(0.8, 0.2)	
		(0.2, 0.5)	
		(0.2, 0.8)	
Right eyebrow		(0.5, 0.2)	0.1
		(0.8, 0.2)	
		(0.2, 0.8)	
		(0.2, 0.5)	
Eyes		(0.2, 0.8)	0.3
		(0.5, 0.5)	
		(0.8, 0)	
		(0.5, 0.2)	
		(0.2, 0.8)	
Face		(0.2, 0.5)	0.1
		(0.5, 0,2)	
Mouth		(0.2, 0.5)	0.4
		(0.2, 0.8)	
		(0.8, 0.2)	
		(0, 0.8)	
		(0.8, 0)	
		(0.2, 0.5)	

3.1.3 Expression feature extraction

After obtaining a cartoon face, we can easily obtain the approximate position of each part to be detected, thereby detecting each bit. Here we adopt the Oriented fast and Rotated Binary robust independent elementary features (ORB) algorithm for feature matching. This algorithm was proposed by Rublee et al.[16] to propose a fast feature point extraction and description algorithm.

The ORB algorithm is divided into two parts, namely, feature point extraction and feature point description. The feature extraction is developed by using the features From Accelerated Segment Test (FAST) algorithm, and the feature point description is improved according to the Binary Robust Independent Elementary Features (BRIEF) feature description algorithm. The ORB feature combines the detection method of FAST feature points with the BRIEF descriptors and then improves and optimizes them based on their originals.

The feature extraction of the ORB algorithm is improved by the FAST algorithm, which then becomes FASTKeypoint Orientation or oFAST. That is to say, after the feature point is extracted using FAST, a feature point direction is defined so that the rotation of the feature point is not deformed. The FAST algorithm is recognized as the fastest feature point extraction method. The feature points extracted by the FAST algorithm are very close to the corner type. The FAST algorithm is as follows:

• Rough extraction extracts a large number of feature points. Specific method: Select a point P from the image as shown in Fig. 4. The method for judging whether the point is a feature point is to draw a circle with a radius of 3 pixels with P as the center. If the gradation value of the consecutive n pixel points on the circumference is larger or smaller than the gradation value of the P point, then P is considered as a feature point.

• Train a decision tree using the ID3 algorithm, and input the n pixels on the circumference of the feature point into the decision tree to filter out the optimal FAST feature points.

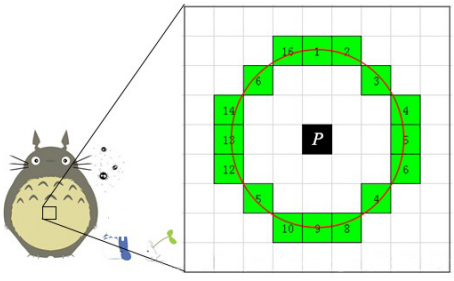

Fig. 4 FAST feature point judgment diagram.

• Non-maximum suppression removes locally dense feature points. Here, the calculation method is the absolute value sum of the feature point P and the deviation of the 16 feature points around it. In comparing the adjacent feature points, the feature points with larger response values are retained and the remaining feature points are deleted.

• Establish pyramids to achieve multiscale invariance of feature points.

• To ensure the rotation invariance of the feature points. The ORB algorithm proposes the use of the moment method to determine the direction of the FAST feature points. This means that the feature point is calculated by the moment with r as the centroid within the radius, and the feature point coordinate to the centroid forms a vector as the direction of the feature point:

$$m_{pq} = \sum_{X,Y \in r} X^p Y^q |(X, Y) \tag{7}$$

where $I(X, Y)$ is the image grayscale expression. The centroid of the moment is given by

$$C = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}} \right) \tag{8}$$

Assuming that the corner coordinate is O, the angle of the vector is the direction of the feature point, which is calculated as follows:

$$\theta = \arctan\left(\frac{m_{01}}{m_{00}} \Big/ \frac{m_{10}}{m_{00}} \right) = \arctan(m_{01}/m_{10}) \tag{9}$$

The ORB uses the BRIEF algorithm to calculate the descriptor of a feature point. The core idea of the BRIEF algorithm is to select N pairs of points around a key point P in a certain pattern and then combine the comparison results of the N point pairs as descriptors.

In the proposed method, based on the above theory, relatively few points are used to extract the facial features. Moreover, the emotional tendencies of each position in the facial features are scored separately by setting different weights (Table 1). The polar emotion scores expressed by facial features are summarized, and then the emotional classification based on facial features is obtained. For example, we extract the left eyebrow from the picture, compare it with the feature model in Table 1, and select the model with the highest similarity. The score of this model is taken as the score of the left eyebrow we extracted, and the other parts are the same. After obtaining the scores of all the parts, we then obtain the sum of the scores of all parts by the

weights of different parts until we finally determine the overall expression score.

3.2 Scene feature extraction

The scene of a cartoon picture is an important basis for analyzing the emotions of cartoon pictures. In our proposed method, we extract the background color of the cartoon picture, convert it into Hue (H), Saturation (S), and Value (V) (HSV) vector, and obtain the polar emotion of the cartoon picture scene using the Adaboost algorithm. The idea of the Adaboost algorithm is to combine the output of multiple Backpropagation (BP) weak classifiers to produce more efficient classification results. The algorithm steps are detailed below.

• Training sample selection and BP network initialization. The δ training data are randomly selected from the sample space, and the distribution weight $D_t(i) = 1/\delta$ of the test data is initialized. Then, the neural network structure is determined according to the sample input and output dimension, and the weight and threshold of the BP neural network are initialized.

• Weak classifier prediction. When training each weak classifier, the predicted output of each BP neural network is obtained, and the prediction error of the prediction sequence $g(t)$ is obtained. The calculation formula is given by

$$\sigma_t = \sum_i D_i(i),\ i = 1, 2, \ldots, \delta(g(t) \neq \eta) \tag{10}$$

where $g(t)$ is the predicted value and η is the expected value.

• Calculate the predicted sequence weights. According to the prediction error a_t of the prediction sequence $g(t)$, the weight a_t of the sequence is calculated. The calculation formula is given by

$$a_t = \frac{1}{2}\ln\left(\frac{1-\sigma_t}{\sigma_t}\right) \tag{11}$$

• Test data weight adjustment. According to the predicted sequence weight a_t, the weight of the next training sample is obtained. The adjustment formula is given by

$$D_{t+1}(i) = \frac{D_t(i)}{B_t} \times \exp[-a_t y_i g(t)],\ i = 1, 2, \ldots, m \tag{12}$$

where B_t is a normalization factor, the purpose of which is to make the distribution weights equal to 1 when the weight ratio is constant.

• Construct a strong classification function. After training T round, the T group weak classification function $f(g(t), a_t)$ is obtained, which is combined to obtain the strong classification function $h(x)$.

$$h(x) = \text{sign}\left[\sum_{i=1}^{T} a_t \cdot f(g(t), a_t)\right] \tag{13}$$

Given that the HSV color space can better reflect the human perception of color, we choose HSV as the workspace. The human visual system is more sensitive to H than S and V, so the HSV space is quantified as follows[17]: H was divided into 10 segments with different lengths, and S and V are divided into three areas: black, gray, and color, where the gray and colored areas required further quantified. The specific division is shown below.

$$H = \begin{cases} 0, & h \in (330, 22]; \\ 1, & h \in (22, 45]; \\ 2, & h \in (45, 70]; \\ 3, & h \in (70, 105]; \\ 4, & h \in (105, 146]; \\ 5, & h \in (146, 167]; \\ 6, & h \in (167, 186]; \\ 7, & h \in (186, 218]; \\ 8, & h \in (218, 278]; \\ 9, & h \in (278, 330], \end{cases}$$

$$S = \begin{cases} 0, & s \in (0.2, 0.65]; \\ 1, & s \in (0.65, 1.0], \end{cases} \tag{14}$$

$$V = \begin{cases} 0, & v \in [0, 0.2]; \\ 1, & v \in (0.2, 0.7]; \\ 2, & v \in (0.7, 1.0] \end{cases}$$

According to the above quantization level, each color component is synthesized into a one-dimensional (1D) feature vector: $l = 6H + 3S + V$. This comprises the HSV color space, which is quantified into 10 kinds of chromaticity, two kinds of saturation, and three kinds of hues, forming a 60-dimensional color eigenvector. Finally, we use the trained Adaboost classifier to classify the picture scene.

3.3 Decision making

Face data. As there may be multiple characters in the picture, we obtain a matrix of $n \times 2$ in the analysis process of the facial part, where n is the number of expressions in the picture. Each row in the matrix is composed of the proportion of the facial expression E and the proportion of the face in the cartoon picture ϵ. Referring to Sentribute in the decision-making method mentioned[18], we redivide the positive, neutral, and negative emotions into $1, 0$, and -1,

respectively, to eliminate the possible impact of the previous step. Through the experiments, we found that according to the approximate proportion of expressions, the weights are assigned, so that the main character expression can have a greater impact on the emotions of the pictures, and thus produce better results in the subsequent classification. After processing, we obtain the total facial expression E_{total}:

$$E_{\text{total}} = \sum \frac{\epsilon_i}{\sum \epsilon_j} E_i \tag{15}$$

E_{total} is an important reference for classification. The positive and negative respectively represent the negative or positive effect of the expression on the emotional tendency of the picture. In addition, ϵ is always used as the reference for the weight of the emotion on the picture.

Data integration. As shown in Fig. 5, before integrating the two pieces of data, we refer to the total detection of facial data and scene data to gain sufficient influence on the emotion of the picture. When its influence on the picture is too small, we ignore it. In this way, we can reduce unnecessary processing time to a certain extent.

Polar emotion classification. After the processes above, we can now obtain a score of picture emotion. We determine the polar emotion classification according to the score, wherein a high-scoring picture is positive and the low-scoring picture is negative.

4 Experimental

In this section, we conduct experiments to validate the proposed method. Due to the limitation of the research in polar emotion classification based on cartoon pictures, there is no unified public data set for our experiments. Based on this problem, we extract and integrate a Moe-style cartoon picture polar emotion classification set from the Danboon2018 (https://www.gwern.net/Danbooru2018) dataset and other open source cartoon picture material libraries. Our dataset contains a total of 6022 pictures.

After determining the dataset set we need, we used tools to preprocess the dataset to obtain the data annotation. In this process, we first used Face++ (https://www.faceplusplus.com.cn) to mark cartoon pictures. Due to the inherent defects of Face++ tools, we used the Labeling tool to continue the data preprocessing of the dataset. The emotion scores and weights used in facial parsing of the proposed are shown in Table 1.

We use precision, recall, and F1-score as the metric to evaluate the performance of the proposed algorithm, and these are defined below.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{16}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{17}$$

$$\text{F1-score} = \frac{2 \times \text{TP}}{2 \times \text{TP} + \text{FP} + \text{FN}} \tag{18}$$

where TP (True Positive) is a positive cartoon picture that is correctly divided into positive emotions, and FP (False Positive) represents a negative cartoon picture that is mistakenly defined as positive emotions, and FN (False Negative) represents a positive cartoon picture that is mistakenly defined as negative emotions.

For comparison, experiments on three baseline method algorithms, Sentribute, CNN, and Pulse Coupled Neural Network (PCNN)[18, 19], respectively, are carried out. All the baseline methods and our methods are run on the dataset above and the result is the average of five times. Specifically, we randomly select four kinds of cartoon pictures to validate our model, and some of them are listed in Fig. 6. From Fig. 6a we can see that the environments are all filled with relatively brighter colors, and the facial

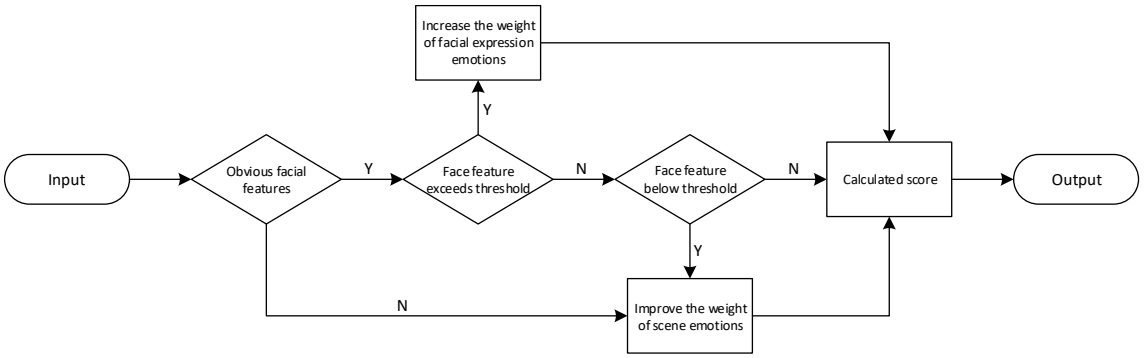

Fig. 5 Data integration flow chart.

Fig. 6 Experimental results: (a) and (b) are positive pictures, (c) and (d) are negative pictures.

expressions are relatively pleasant and warm, which are in harmony with the environment. In the case of positive people with a positive environment, our model judges them as positive pictures. From Fig. 6b, we can see that the environments are also filled with relatively brighter colors; although the facial expressions are relatively distressed, the overall image is positive and there are different facial emotion features within that environment. Negative people with a positive environment. However, because the proportion of the scene is higher, the negative score of the expression is not high, so our model gives a positive judgment during the analysis. From Fig. 6c, we can see that the environments are also filled with relatively darker colors, whereas the facial expressions are relatively distressed. In the case of positive people with a negative environment, similar to the above, we make negative judgments based on the scene when the expression is intense and the degree of influence is not high. From Fig. 6d, we can see that the environments are also filled with relatively darker colors, whereas the facial expressions are relatively pleasant. In the case of negative people with a negative environment, our model judges them as negative pictures.

The simulation results are shown in Tables 2 and 3. From the results, we can see that the proposed algorithm shows better performance in determining the emotion features of cartoon pictures compared with others.

The average accuracy in determining the emotion feature of cartoon pictures demonstrates that CNN has the lowest accuracy, Sentribute performs well in some of the image, and PCNN has better performance in our dataset. However, the proposed algorithm obtains a high classification accuracy.

Table 2 Average accuracy comparison.

Algorithm	Precision	Recall	F1-score
Ours	0.828	0.814	0.819
Sentribute[18]	0.801	0.792	0.796
CNN[19]	0.796	0.806	0.801
PCNN[19]	0.813	0.812	0.818

In order to well analyze the performance of the proposed method, we add the comparable experiments with Sentribute. The results show that performance is less than that of Sentribute in determining the emotion feature of environment, although the proposed method performed well in facial emotion recognition. Meanwhile, benefiting from the combination of facial and environment emotion features, the final accuracy is higher than that of Sentribute in the cartoon picture dataset.

In summary, from the evaluation of our algorithm, we can see that benefiting from the combination of facial and environment emotion features, the proposed method shows a competitive performance when compared with other algorithms.

Our method is worse than Sentribute when it comes to scene analysis, but the former is much better when we analyze facial expressions, as shown in how we model the face of the cartoon characters. Combining the results of scene and the face, our method obtained better results and showed 2% improvement in the overall

Table 3 Accuracy comparison of the proposed with Sentribute.

Algorithm	Emotion figure		Facial emotion		Environment emotion	
	Precision	Recall	Precision	Recall	Precision	Recall
Ours	0.828	0.814	0.898	0.88	0.636	0.627
Sentribute[18]	0.801	0.792	0.718	0.677	0.828	0.803

effect.

5 Conclusion

The sentiment analysis for cartoon material helps speed up the cartoon enterprise's screening process for suitable cartoon animation materials. However, there are relatively few studies on emotion recognition for cartoon materials. Therefore, in this paper, we design a polar emotion classification algorithm based on image sentiment, which is suitable for cartoons considering its unique characteristics. In our polar sentiment classification algorithm, we consider two main issues: facial feature extraction and scene feature extraction. Based on the characteristics of the Moe drawing style according to the related knowledge of existing comics and cartoons, our algorithm can model the face of the Moe-style cartoon character, use five feature points to express the position of the facial features in a simple way, and design the algorithm model for the five senses. We can obtain the emotional tendency of cartoon faces by features matching. Secondly, the scene of cartoon pictures is also used as the basis of sentiment analysis. Finally, we can determine the polar emotion of cartoon pictures based on the results of expression analysis and scene analysis. We designed a cartoon picture dataset and compared it with Sentribute, CNN, and PCNN. The experimental results show that our method has certain competitiveness in the classification of polar emotions in cartoon pictures. In our future work, we plan to extend polar sentiment classification to multiple emotions classification and optimize the existing methods to increase efficiency.

Acknowledgment

This work was supported by the National Key Research and Development Plan of China (No. 2017YFD0400101). We would like to gratefully acknowledge Mr. Jianbo Yuan as well as Ms. Gwern Branwen for making the model and datasets available.

References

[1] J. Huo, W. B. Li, Y. H. Shi, Y. Gao, and H. J. Yin, WebCaricature: A benchmark for caricature recognition, arXiv preprint: 1703.03230, 2017.

[2] B. F. Klare, S. S. Bucak, A. K. Jain, and T. Akgul, Towards automated caricature recognition, in *2012 5^{th} IAPR_Int. Conf. on Biometrics*, New Delhi, India, 2012, pp. 139–146.

[3] S. X. Ouyang, T. Hospedales, Y. Z. Song, and X. M. Li, Cross-modal face matching: Beyond viewed sketches, in *Computer Vision*, D. Cremers, I. Reid, H. Saito, and M. H. Yang, eds. Springer, 2015, pp. 210–225.

[4] S. Siersdorfer, E. Minack, F. Deng, and J. Hare, Analyzing and predicting sentiment of images on the social web, in *Proc. 18^{th} ACM Int. Conf. on Multimedia*, Firenze, Italy, 2010, pp. 715–718.

[5] B. Li, S. H. Feng, W. H. Xiong, and W. M. Hu, Scaring or pleasing: Exploit emotional impact of an image, in *Proc. 20^{th} ACM Int. Conf. on Multimedia*, Nara, Japan, 2012, pp. 1365–1366.

[6] D. Borth, R. R. Ji, T. Chen, T. Breuel, and S. F. Chang, Large-scale visual sentiment ontology and detectors using adjective noun pairs, in *Proc. 21^{st} ACM Int. Conf. on Multimedia*, Barcelona, Spain, 2013, pp. 223–232.

[7] C. Xu, S. Cetintas, K. C. Lee, and L. J. Li, Visual sentiment prediction with deep convolutional neural networks, arXiv preprint: 1411.5731, 2014.

[8] V. Campos, B. Jou, and X. Giró-i-Nieto, From pixels to sentiment: Fine-tuning CNNs for visual sentiment prediction, *Image and Vision Computing*, vol. 65, pp. 15–22.

[9] C. de Juan and B. Bodenheimer, Cartoon textures, in *Proc. 2004 ACM SIGGRAPH/Eurographics Symp. on Computer Animation*, Grenoble, France, 2004, pp. 267–276.

[10] H. H. Wang and B. Raj, On the origin of deep learning, arXiv preprint: 1702.07800, 2017.

[11] H. S. Bhatt, S. Bharadwaj, R. Singh, and M. Vatsa, Memetically optimized MCWLD for matching sketches with digital face images, *IEEE Trans. Inform. Forensics Secur.*, vol. 7, no. 5, pp. 1522–1535, 2012.

[12] A. Ruiz-Garcia, M. Elshaw, A. Altahhan, and V. Palade, Deep learning for emotion recognition in faces, in *Int. Conf. on Artificial Neural Networks 2016*, A. Villa, P. Masulli, and R. A. Pons, eds. Springer, 2016, pp. 38–46.

[13] B. Abaci and T. Akgul, Matching caricatures to photographs, *Signal Image Video Process.*, vol. 9, no. S1, pp. 295–303, 2015.

[14] E. J. Crowley, O. M. Parkhi, and A. Zisserman, Face painting: Querying art with photos, in *British Machine Vision Conf.*, Swansea, UK, 2015, pp. 1–13.

[15] K. P. Zhang, Z. P. Zhang, Z. F. Li, and Y. Qiao, Joint face detection and alignment using multitask cascaded convolutional networks, *IEEE Signal Process. Lett.*, vol. 23, no. 10, pp. 1499–1503, 2016.

[16] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, ORB: An efficient alternative to SIFT or SURF, in *2011 Int. Conf. on Computer Vision*, Barcelona, Spain, 2012, pp. 2564–2571.

[17] J. F. Cao, J. J. Chen, and H. F. Li, Sentiment classification of image based on Adaboost-BP neural network, (in Chinese), *J. Shanxi Univ. (Nat. Sci. Ed.)*, vol. 36, no. 3, pp. 331–337, 2013.

[18] J. B. Yuan, S. Mcdonough, Q. Z. You, and J. B. Luo, Sentribute: Image sentiment analysis from a mid-level perspective, in *Proc. 2^{nd} Int. Workshop on Issues of Sentiment Discovery and Opinion Mining*, Chicago, IL, USA, 2013.

[19] Q. Z. You, J. B. Luo, H. L. Jin, and J. C. Yang, Robust image sentiment analysis using progressively trained and domain transferred deep networks, in *Proc. 29^{th} AAAI Conf. on Artificial Intelligence*, Austin, TX, USA, 2015, pp. 381–388.

Qinchen Cao is a master student with the School of Computer Engineering and Science, Shanghai University, China. His research interests include deep learning and computer vision.

Weilin Zhang is a master student with the School of Computer Engineering and Science, Shanghai University, China. His research interests include information retrieval and recommender systems.

Yonghua Zhu is an associate professor at Shanghai Film Academy in Shanghai University. He received the PhD degree from Shanghai University in 2006. His current research interests include artificial intelligence and Internet of Things.