CNN Models for Classifying Emotions Evoked by Paintings

Wanliang Tan, Jiahui Wang, Yu Wang
Stanford University
wanliang, jiahuiw, yuwangme@stanford.edu

Mana Lewis*
Chez Mana Co.
mana@chezmana.com

William Jarrold*
UC Davis
william.jarrold@gmail.com

Abstract

Image-related classification has been extensively studied and implemented in the past few years. However, human emotions evoked by paintings created by artists with different styles can be much harder to grasp and analyze. Thus it is interesting to implement a painting classifier to learn the painting features and label the emotions they evoke in humans in an automated fashion. In this paper, we trained two CNN models, based on VGGnet-16 and ResNet-50, on a collection of 3104 paintings with human labeled emotions. The training results of those two models are discussed and compared.

1. Introduction

Nowadays, art paintings are playing a more and more important role in everyone's daily life. From art gallery collection to paintings used by advertisement companies to grab attention and market products, paintings are found everywhere.

As a mode of creative expression, paintings can consists of many artistic objects and techniques including drawing, gesture, composition, narration, abstraction[1]. They can be naturalistic and representational, photographic, abstract, symbolic or emotive in nature. When we see these paintings with different styles, we feel different emotions such as happy or sad, peaceful or angry. The compositions, color tones, brush strokes affect our feelings in many mysterious ways.

To solve this mystery, we use Convolutional Neural Networks (CNN) to build classification models for emotions evoked by these paintings. CNN are a class of machine learning architecture that perform especially well at learning complex target function. In particular, they have been implemented extremely successfully to analyze visual images. In this paper, we will mainly use transfer learning and build up our models based on VGGnet-16[1] and ResNet-50[2], which contains a combination of convolutional and fully connected layers and won ImageNet Large Scale Visual Recognition Challenge (ILSVRC)[3] in 2014 and 2015 respectively.

*Mana and William generously provided us the dataset and are not enrolled in CS 231N

[1]https://en.wikipedia.org/wiki/Painting

2. Related Work

Recently there is a growing interest from various research communities in understanding the emotional response of the viewer during interaction with artworks. A psychological study on the effects of colors on emotions based on Pleasure, Arousal and Dominance model shows that brighter colors are more pleasant, less arousing, and induce less dominance than the darker colors[4]. In [5], researchers used factor analysis method and investigated how eleven emotion scales are associated with three color-emotion factors (i.e., color activity, color weight and color heat) of single colors, which shows that there is consistency in the way people perceive colors.

To computationally solve this problem, researchers have done a lot of works on this. In [6], they used Supporting Vector Machines to assess the local image statistics. Sartori et al[7] proposed to use both visual and text information in a joint learning model for abstract painting emotion recognition. Liu et al.[8] used a multi-task learning approach for painting style analysis. These models are all traditional statistic models and don't apply deep neural networks.

Since 2012[3], deep learning has made significant advances in Computer vision tasks. As a result, higher-level visual semantics such as image aesthetic analysis[9] and visual sentiment analysis[10] are becoming more and more tractable. You et al.[11] used CNN to learn features which are useful for visual analysis. And then they[12] proposed a cross-modality consistent regression (CCR) scheme for joint textual-visual sentiment analysis, which achieved the best performance over other fusion models. These methods are all legitimate except that the neural networks are not deep compared to VGGnet-16 and ResNet-50.

There have been several papers which apply state-of-art CNN to perform emotion classification on images. In [13], they applied VGGnet-16 and proposed several strategies of generating visual attributes. As a result, they

achieved about 70 % accuracy for the final classifier. Gajarla et al.[14] experimented with various classification methods on data from Flickr - SVM on high level features of VGG-ImageNet, fine-tuning on pretrained models like RESNET, Places205-VGG16 and VGGImageNet. In [15], they achieved accuracy improvement for emotion identification by fine-tuning a CaffeNet CNN architecture.

What we are trying to do here is different from them in that the data we used is different. We are trying to use CNN to identify emotions evoked by viewing images of artwork from Wikiart. These images are drawn by different artists with distinct styles, which makes the labeling and modeling not easy as we will discuss in the following sections.

3. Problem Statement

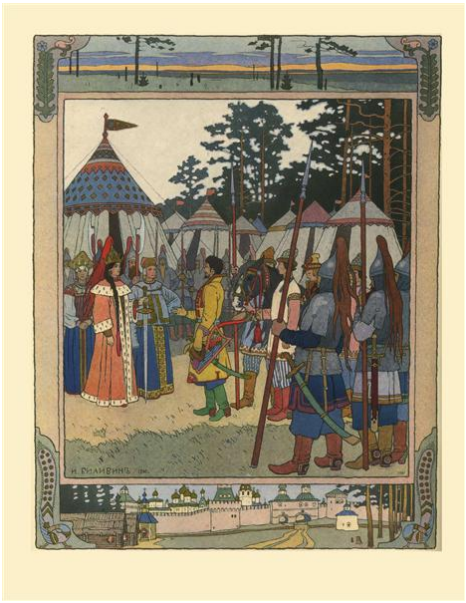

Figure 1. Example of a painting: Illustration for the Russian Fairy Story "Maria Morevna". Tagged as knights-and-warriors with emotion "optimism".

We use a dataset of 3104 paintings, each has a theme tag and an emotion label. One example of the paintings in the dataset is shown in Figure 1[2]. This painting is available on `WikiArt.org`, which is a non-profit online home for visual arts from all around the world. Most images on WikiArt has a series of metadata including English title, original title, date, style, series, genre, and tags. The first tag from the website is given as the theme tag mentioned above. The associated emotion is labeled manually by human experts. Figure 1 is tagged as "knights-and-warriors" and its associated emotion is "optimism". The emotion labels are taken as given, though sometimes it is not completely clear as to why a painting has certain emotion label.

The 3104 paintings have different pixel size. The average size is 922x890. To allow for VCCnet-6 and ResNet-50 training and speed up the computation, the images are shrinked to 224x224, which is the same size as images in the Cifar-10 dataset. The resize distorts the paintings a little due to change in aspect ratio. This distortion is neglected in this paper.

The goal of the project is to identify the emotions in images of paintings and identify the explanation for those feelings. We expect the trained CNN models to correctly predict the emotion label with accuracy over 60%. The accuracy is defined to be the portion of paintings with correctly labeled emotions. This seemingly low expectation reflects our reservation on CNN's capability of predicting artistic objects and irregular images.

4. Technical Approach

Since our dataset only contains 3104 paintings, we chose to use transfer learning and fine tune ImageNet-pretrained VCCnet-16 and ResNet-50 models with pretrained weights. We used Keras as high level wrapper with TensorFlow as backend. In addition, we implemented simple CNN as a baseline model.

4.1. Simple CNN

As a benchmark, we implement a simple CNN model using two-layer constitutional networks with maxpooling and dropout as one block. By changing the block numbers and tuning parameters, the best test accuracy we can get is 41% as shown in Table 3.

4.2. VGGnet-16

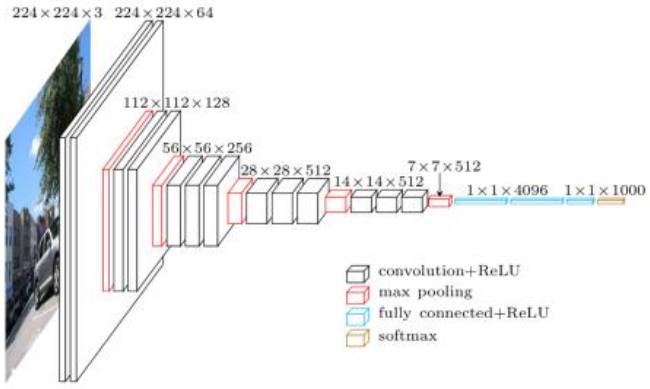

Figure 2. VGGNet-16 architecture

The configuration of VGGnet 16[3] is shown in the pic-

[2]https://www.wikiart.org/en/ivan-bilibin/illustration-for-the-russian-fairy-story-maria-morevna-1900-3

[3]https://blog.heuritech.com/2016/02/29/a-brief-report-of-the-heuritech-deep-learning-meetup-5/. 2

ture above. We used Keras.applications.VGG16 class as our training model. Overall we kept the majority of the bottom structure of VGG16. The convolutional layers were initialized with weights pretrained on ImageNet dataset. We did not include the top fully-connected layers in VGG16. Instead, we added two new fully connected layers with randomly initialized weights and 11 final outputs. We froze the bottom convolutional layers and only trained the top fully connected layers, which is the essence of transfer learning. The list of parameters we used for the best models in Table 3 are shown below:

- learning rate: 1×10^{-5}
- mini-batch size: 64 (we used small size to avoid resource exhausted error)
- learning rate decay: 1×10^{-8}
- dropout with keep_prob = 0.3
- loss function: sparse categorical loss

The optimizer was chosen to be Adam. Adam combines the benefits of Stochastic Gradient Descent (SGD) and learning rate annealing. SGD estimates the gradient of loss function using a small batch of data and iteratively update the model weights to minimize the loss. Momentum 0.9 accelerates iteration convergence by accumulating pass gradient knowledge to overcome vanishing gradient.

4.3. ResNet-50

In this part, we chose ResNet-50 as another starting point for CNN model for its deep architecture. A schematic of its architecture[4] is shown in Figure 3. We successfully built and started training an ImageNet pre-trained ResNet-50 model based on codes from [5]. We changed the last fully connected layer to output 11 scores which then transfer to classification probabilities through softmax function. We used cross entropy loss and defined the prediction accuracy to be the fraction of correct predictions for the test set.

In terms of the model hyper-parameters, we closely followed recommendations from [6]. The list below shows a summary of the hyper-parameters we used in the best models in Table 3.

- learning rate: 0.1
- mini-batch size: 80 (we used small size to avoid resource exhausted error)
- learning rate decay: 1×10^{-5}

[4]CS 231n lecture notes
[5]https://github.com/flyyufelix/cnn_finetune/blob/master/resnet_50.py
[6]http://cs231n.stanford.edu/slides/2018/cs231n_2018_lecture09.pdf

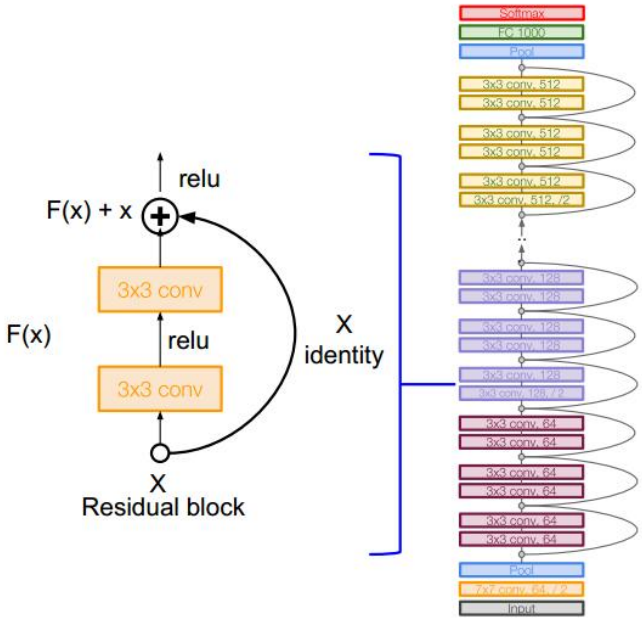

Figure 3. ResNet architecture

- L2 regularization on all convolution layer and dense layer weights with $\lambda = 0.005$
- no dropout
- loss function: sparse categorical loss

The optimizer is chosen to be SGD with Momentum 0.9, which is the recommended one in [7].

5. Experiments

5.1. Trainable layers number, L2-regularization, dropout

Overfitting is a major problem during our model tuning so a signification portion of time was spent on implementing and tuning different model regularization techniques. These techniques include reducing number of trainable parameters, adding L2-regularization, and increasing dropout probability.

The number of trainable parameters is a tunable hyper-parameter since we are using transfer learning. We trained each model with different number of trainable parameters. We observed significant gap between training and testing accuracy which indicates overfitting when too many layers are allowed to update. Decreasing the number of trainable parameters significantly reduces overfitting but also limits model capacity and slightly reduces test accuracy. It is currently our major way of regularization. For all models we

[7]http://cs231n.stanford.edu/slides/2018/cs231n_2018_lecture09.pdf

reported test accuracy in section 6, only the last one or two dense layers are allowed to update during training.

We also tried different L2-regularization. Although increasing L2-regularization reduces overfitting, it also reduces test accuracy to such extend that VGGnet-16 and ResNet-50 underperform simple CNN baseline. We also examined the final accuracy of the model with dropout. To use VGGnet-16 as an example, the final accuracy is 50% for test set without dropout. If set dropout keep_prob to be 0.3, which means the probability of an element to be kept is 0.3, the final test accuracy is 53%. Adding dropout seems to be of some help. But considering that 70% of dense layer nodes are dropped out and the improvement is only 3%, it is difficult to draw a definite conclusion.

5.2. Dataset modification

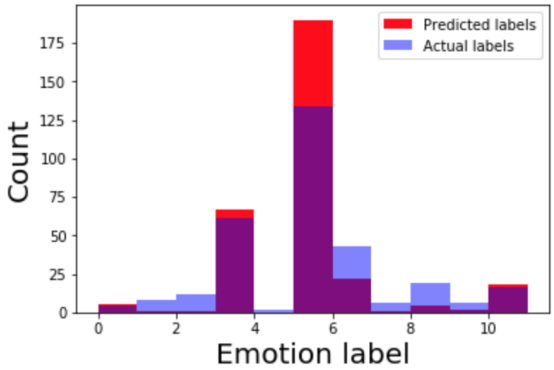

Figure 4. Count of predicted and actual labels for test dataset

After 5 epochs of training, both VGGnet-16 ResNet-50 model is able to achieve about 50% test accuracy. Two overlapping histograms of predicted and actual labels for test dataset (310 paintings) are shown in Figure 4. These two histograms show the comparison between data and prediction distribution. The mapping from label number to emotion label is shown in Table 1. The model tends to predict too many paintings as "neutral" (label 5), which represents over 42% of all 3104 paintings. The over-prediction is likely due to the obvious data imbalance in this dataset, as shown in Figure 5. Except for "neutral" and "joy", most of the other labels are underrepresented in the dataset.

Table 1. Label number to emotion label mapping

Number	0	1	2	3	4
Emotion	fear	disgust	surprise	optimism	envy
5	6	7	8	9	10
neutral	joy	anger	sadness	lust	love

To alleviate the data imbalance, we decided to modify the labels of the paintings with the help from Mana. We

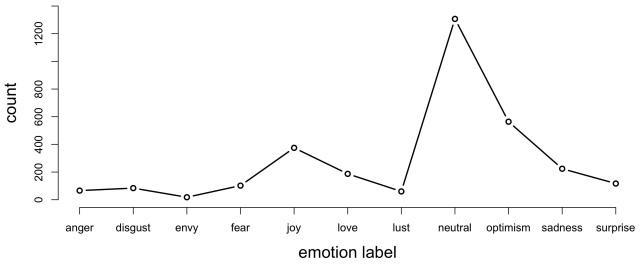

Figure 5. Data imbalance

grouped joy, love, optimism and surprise as POSITIVE; anger, disgust, envy, fear, lust and sadness as NEGATIVE; the all the rest as NEUTRAL. The neutral set is actually unchanged. By modifying the dataset this way, it becomes much more balanced: 1243 positive, 554 negative and 1307 neutral. The old and new label sets are shown in Table 1 and Table 2.

Table 2. Grouped 3 labels

Number	0	1	2
Emotion	Positive	Negative	Neutral

As an empirical results, grouping labels to reduce data imbalance helps improve training and testing accuracy.

5.3. Data Generation

We used keras image processing package *ImageDataGenerator* to generate more image to make the data balanced. The data augmentation methods include rotation, scaling flipping and translation. For 3-label grouped datasets, we augmented figures with "Positive" label to 2091, with "Neutral" label to 2113, and with "Negative" label to 2219. However, the augmentation did not help a lot in improving the test accuracy. The highest test accuracy we was 46%.

5.4. Others

We also adjusted other parameters such as batch size, loss function, learning rate, optimization algorithm to improve the training results. In general these parameters do not affect training and testing accuracy significantly, given that they are in a reasonable range. Although batch size does not affect the model tuning much, it together with the number of trainable parameters are limited by GPU memory, which is 12GB in case of NVIDIA K80. We also experimented with mean-square loss and cross entropy loss with one-hot class labels, sparse cross entropy with categorical class labels. The training results are similar. Each CNN architecture is optimized with SGD and Adam algorithm. The training process is again similar.

6. Results

Regularization proved to be one of the most significant factor in this project. For every model that was implemented, significant overfitting could easily occur without carefully selected regularization.

6.1. Best model accuracy

Table 3. Comparison of the accuracy of three architectures with 3 label dataset

	Simple CNN	VGGnet-16	ResNet-50
Best model training accuracy	51%	98%	94%
Best model testing accuracy	41%	56%	53%

The training and testing accuracy for the best model of the three architectures above with 3 labels (section 4) are shown in Table 3. As can be seen from the table, even with all the regularization techniques we implemented, VGGnet-16 and ResNet-50 still have big gaps between training and testing accuracy. This could be due to that model bias instead of variance is the main reason for this gap. In other words, CNN models like VGGnet-16 or ResNet-50 may have strong biases with respect to the artistic painting dataset and the artificial emotion labeling.

The typical accuracy and loss curve during training are shown in Figure 6 and Figure 7.

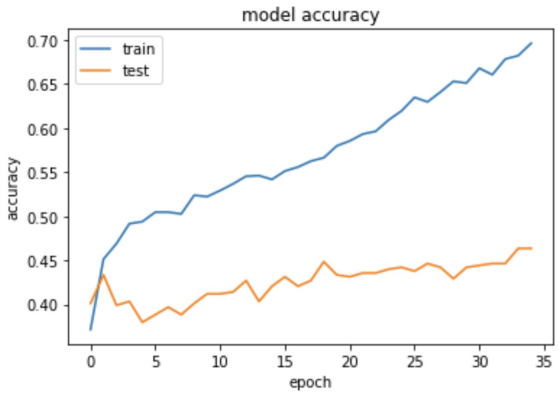

Figure 6. Training accuracy curve

6.2. Visual interpretations

In this section, three types of classification results of VGGnet-16 with 3 grouped labels are presented: explainable correct prediction, explainable wrong prediction, and unexplainable prediction (correct or wrong). For all figures shown below, the vertical label is the true label by artists and the horizontal label is VGGnet-16 prediction.

Figure 8 shows two representative correct predictions. In our painting dataset, ships on water and town overlook are

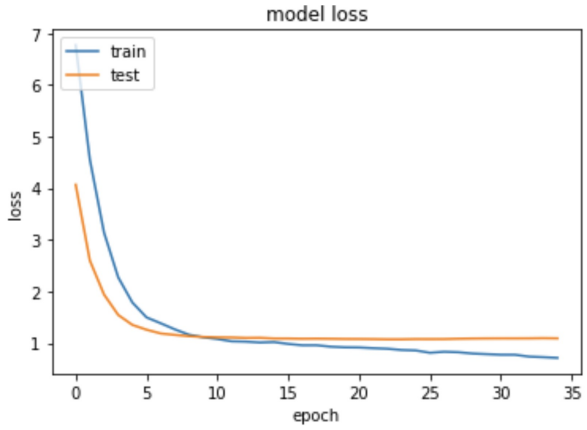

Figure 7. Loss curve

Figure 8. Explainable correct predictions

two common scenes and are usually labeled ”neutral” or ”positive”. The CNN models correctly classify them.

Figure 9. Explainable wrong predictions

In Figure 9 there are two wrong predictions. The left figure shows a violent scene where a group of people attacks a person in the center. Although the true label correctly reflects the evoked negative emotion, the model classifies it as positive. This is a typical example of the model confusing ”positive” and ”negative” as mentioned in subsection 6.3. In this particular case, it is likely due to the warm tone of the painting. The content of the right painting is a typical town overlook scene, which is labeled ”neutral” in the dataset but predicted as ”positive” by the model. For a lack of better interpretation, it is suspected that the green and springtime

color tone is partially responsible for the misclassification.

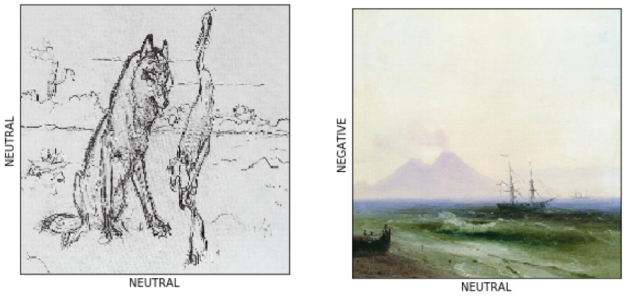

Figure 10. Confusing predictions

Figure 10 presents two confusing predictions. The left one is a correct "positive" prediction of a painting depicting a wolve watching a bird. The true label and the prediction are both "neutral" even though the painting may likely evoke a negative feeling given its depiction of a predator watching its prey and the potential tension. The right figure is a wrong prediction of a common scene of a ship on water. Although the "neutral" prediction seems consist with other painting depicting the similar scenes, the real label is "negative". The negativity for human labelers is suspected to come from the volcano eruption in the background. It is possible that the model can not pay attention to the eruption since water makes the majority of meaningful contents.

6.3. Confusion matrix

As a summary of the prediction accuracy breakup, the confusion matrix of 3-label VGGnet-16 prediction results is shown in Figure 11. In this matrix, x labels are the true labels and y are the predictions. The model tends to predict paintings with "positive" labels as "negative". Empirically this is a major type of misclassification.

7. Conclusion

We trained and fine-tuned three CNN models with the original and the grouped painting datasets. The highest final test accuracy among all models is 56%, reached by training VGGnet-16 with the grouped dataset. Although it does not exceed our original expectation 60%, this accuracy should reflect the capability of moderately tuned VGGnet-16 and ResNet-50 model. As mentioned in section 3, we have reservations regarding the prediction capability of CNN models on paintings due to their artistic and underdefined nature. It may be more difficult for CNN models to extract features from paintings than from more well-defined datasets like ImageNet.

In addition, the true labels may contain noise that highly depends on labeler. To be specific, if the labeling process is viewed as a time series, labels generated by one artist may

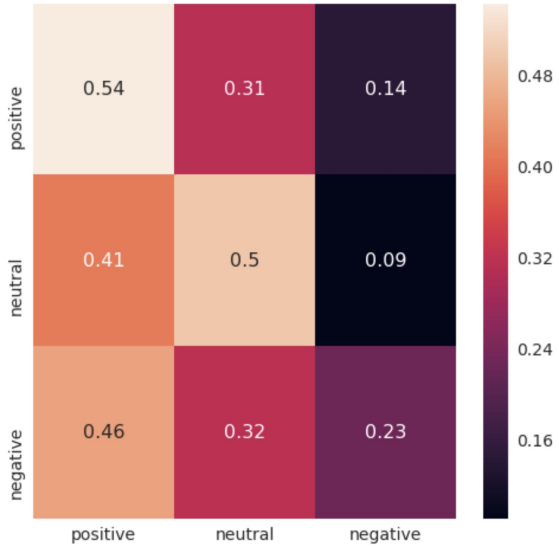

Figure 11. Confusion matrix. x: true label; y: prediction

be autocorrelated due to accumulating impact of the paintings on the emotion of the artist. This autocorrelation is in some sense unaccounted error in the dataset and the model, which could confuse the CNN models. Besides, as discussed in subsection 6.2, part of the true labels provided can be difficult to be interpreted with consensus by human since people respond differently to stroke edges, color tones, and objects of paintings. All these potential issues with true labels raise the question about the suitability of painting-evoked emotion prediction with CNN models. For future work, it may be worthwhile to consider boosting CNN models with other models such as SVM, nearest neighbors.

8. Contributions & Acknowledgements

Y. W., W. T. and J. W. designed, implemented and trained the neural networks. Y. W., W. T. and J. W. wrote the paper. M. L. and W. J. provided experimental data and advised on the project.

References

[1] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[2] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

[3] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, pp. 1097–1105, 2012.

[4] P. Valdez and A. Mehrabian, "Effects of color on emotions.," *Journal of experimental psychology: General*, vol. 123, no. 4, p. 394, 1994.

[5] L.-C. Ou, M. R. Luo, A. Woodcock, and A. Wright, "A study of colour emotion and colour preference. part i: Colour emotions for single colours," *Color Research & Application*, vol. 29, no. 3, pp. 232–240, 2004.

[6] V. Yanulevskaya, J. C. van Gemert, K. Roth, A.-K. Herbold, N. Sebe, and J.-M. Geusebroek, "Emotional valence categorization using holistic image features," in *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*, pp. 101–104, IEEE, 2008.

[7] A. Sartori, Y. Yan, G. Özbal, A. A. A. Salah, A. A. Salah, N. Sebe, *et al.*, "Looking at mondrian's victory boogie-woogie: what do i feel?," in *IJCAI*, vol. 1, p. 3, 2015.

[8] G. Liu, Y. Yan, E. Ricci, Y. Yang, Y. Han, S. Winkler, N. Sebe, *et al.*, "Inferring painting style with multi-task dictionary learning.," in *IJCAI*, pp. 2162–2168, 2015.

[9] X. Lu, Z. Lin, H. Jin, J. Yang, and J. Z. Wang, "Rapid: Rating pictorial aesthetics using deep learning," in *Proceedings of the 22nd ACM international conference on Multimedia*, pp. 457–466, ACM, 2014.

[10] D. Borth, R. Ji, T. Chen, T. Breuel, and S.-F. Chang, "Large-scale visual sentiment ontology and detectors using adjective noun pairs," in *Proceedings of the 21st ACM international conference on Multimedia*, pp. 223–232, ACM, 2013.

[11] Q. You, J. Luo, H. Jin, and J. Yang, "Robust image sentiment analysis using progressively trained and domain transferred deep networks.," in *AAAI*, pp. 381–388, 2015.

[12] Q. You, J. Luo, H. Jin, and J. Yang, "Cross-modality consistent regression for joint visual-textual sentiment analysis of social multimedia," in *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining*, pp. 13–22, ACM, 2016.

[13] Q. You, H. Jin, and J. Luo, "Visual sentiment analysis by attending on local image regions.," in *AAAI*, pp. 231–237, 2017.

[14] V. Gajarla and A. Gupta, "Emotion detection and sentiment analysis of images," *Georgia Institute of Technology*, 2015.

[15] V. Campos, B. Jou, and X. Giro-i Nieto, "From pixels to sentiment: Fine-tuning cnns for visual sentiment prediction," *Image and Vision Computing*, vol. 65, pp. 15–22, 2017.