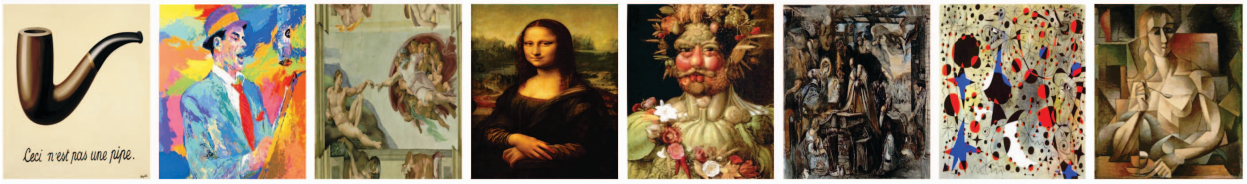
Ceci n'est pas une pipe: A Deep Convolutional Network for Fine-art Paintings Classification

Wei Ren Tan*, Chee Seng Chan†, Hernán E. Aguirre*, and Kiyoshi Tanaka*
*Faculty of Engineering, Shinshu University, Nagano, Japan
†Center of Image and Signal Processing, Faculty of Computer Science & Information Technology,
University of Malaya, Kuala Lumpur, Malaysia
Email: {14st203c; ahernan; ktanaka}@shinshu-u.ac.jp, cs.chan@um.edu.my



***Abstract*—"*Ceci n'est pas une pipe*" French for "This is not a pipe". This is the description painted on the first painting in the figure above. But to most of us, how could this painting is not a pipe, at least not to the great Belgian surrealist artist René Magritte. He said that the painting is not a pipe, but rather an image of a pipe. In this paper, we present a study on large-scale classification of fine-art paintings using the Deep Convolutional Network. Our objectives are two-folds. On one hand, we would like to train an end-to-end deep convolution model to investigate the capability of the deep model in fine-art painting classification problem. On the other hand, we argue that classification of fine-art collections is a more challenging problem in comparison to objects or face recognition. This is because some of the artworks are non-representational nor figurative, and might requires imagination to recognize them. Hence, a question arose is that does a machine have or able to capture "imagination" in paintings? One way to find out is train a deep model and then visualize the low-level to high-level features learnt. In the experiment, we employed the recently publicly available large-scale "Wikiart paintings" dataset that consists of more than 80,000 paintings and our solution achieved state-of-the-art results (68%) in overall performance.**

I. Introduction

Recent years, due to the rapid advancement of digital acquisition of fine-art paintings, vast digital collections have been made available across the Internet and museums. With such a massive digital artwork collections, automated paintings analysis has become an important task in assisting the curators in their daily work routine, such as forgery detection [1], objects retrieval [2], [3], archiving and retrieving artworks [4], etc. More interestingly, a recent work found out that neural algorithm can reconstruct the painting style of different artists [5].

While most of the research has been focusing on object recognition in natural images [6]–[11], little attention was given to the classification of fine-art paintings. One of the key factors is the availability of a decent fine-art paintings dataset for such an evaluation. As evidence, in the object recognition, there are PASCAL VOC [12], CIFAR-10 [13] and ImageNet [14] datasets; while in the scene recognition, there is Places dataset [15] that contain ten thousands to millions of images. Contrary, only a few, very small paintings datasets have been made publicly available. For instance, Khan et al. [16] proposed a dataset consists of 4,266 paintings only. Whereas, dataset used in [17]–[20] have less than 1,000 paintings. Until recently, [21] provided a new dataset, namely the Wikiart paintings dataset[1] that consists of more than 80,000 of paintings.

In this paper, we present a study on large-scale *style, genre*, and *artist* classification of fine-art paintings using the Wikiart paintings dataset with Convolutional Neural Network (CNN). Our objectives to employ the CNN are two-folds. On one hand, we would like to train an end-to-end CNN model to investigate the capability of the CNN in fine-art paintings classification problem. This is in contrast to [21], [22], where the authors investigated the effects of different features coupled with different type of metrics to perform the paintings classification task. Although the CNN was employed, it was simply used as a feature extractor only.

On the other hand, we argue that classification of fine-art collections is a more challenging problem, in comparison to recognizing objects such as dog or cat; or human faces. In general, individuals could differentiate simple paintings categories, for instance, Portrait or Landscape. However, one will require strong background in the art domain (i.e the history) for more complex paintings classification, such as the Abstract and Illustration paintings, as these kinds of artworks are non-representational nor figurative, and might require imagination to recognize them. For example, the second last painting in the figure above, namely *"The nightingale's song at midnight and morning rain"* is a piece of the 23 small paintings on paper (Constellations series), initiated by the great artist Joan Miró

[1]Paintings are collected from http://www.wikiart.org/

in 1939, belongs to this category. The Constellations is Miró's most luminous and affecting series of painting as it captures and represents the most vibrant expressions of Miró's inner universe during the outbreak of the Second World War. He explained the genesis in a letter to a friend: *"I had always enjoyed looking out of the windows at night and seeing the sky and the stars and the moon, but now we weren't allowed to do this any more, so I painted the windows blue and I took my brushes and paint, and that was the beginning of the Constellations."* Hence, a question arose is that does a machine have or able to capture "imagination" in paintings? One way to find out is to train a CNN and then visualizes the low to high-level features learnt by the CNN [23].

As a summary, the contributions of this paper are as follows: First, we pre-trained the CNN using ImageNet and tested the transferability of the learnt features to paintings classification. It has been proven that transferring the well-learnt knowledge from one *source* to a *target* domain improves the deep model accuracy significantly [24]–[27]. During the transfer learning, a new softmax layer replaces the last layer of the pre-trained CNN. In conjunction, we explored two different configurations: a) train the new softmax layer only; b) train the new softmax layer and fine-tune the lower layers at the same time. We found out that the latter strategy performs the best and achieved state-of-the-art results (68%), in comparison to [21] (56%) in the Wikiart paintings dataset. Secondly, we visualize the extracted features of the trained CNN by picturing the responses of neurons. The visualizations show that those features extracted from the paintings of the same group could variant greatly, in contrast to object or face recognition where features extracted from the same class is somehow similar. Therefore, it shows that the fine-art paintings recognition is more challenging.

The rest of the paper is organized as follows: Section II provides a brief introduction of the Wikiart paintings dataset. Section III describes the architecture of the CNN employed in this paper. Experimental results and discussions are presented in Section IV. Finally, conclusions are drawn in Section V.

II. Wikiart Paintings Dataset

The Wikiart paintings dataset [21] has a collection of more than 80,000 fine-art paintings from more than 1,000 artists, ranging from fifteen century to modern times. This dataset contains 27 different styles and 45 different genres. Based on our knowledge, it is currently the largest digital art datasets available publicly. However, not all paintings are included in the classification tasks due to limited number of samples available in some classes for the tasks. To be specific, all paintings are used for *style* classification. Meanwhile, only 10 genres with more than 1,500 paintings are chosen for *genre* classification, with a total of around 65,000 samples. Similarly, only a subset of 23 artists with more than 500 paintings is chosen, with total of around 20,000 images for *artist* classification. Due to space constraint, detailed list of the styles, genres, artists in the dataset, and sample of the digital collections are presented in the supplementary material.

III. Convolutional Neural Network

The overall structure of our CNN has five convolutional layers (conv1-5), three max-pooling layers (max1-3), and three fully connected layers (fc6-8). The design of our CNN architecture is inspired by the AlexNet [28]. The input of the network is $227 \times 227 \times 3$ paintings image. Each convolutional layer yields 96, 256, 384, 384, and 256 feature maps, respectively. The number of neurons in fc6 and fc7 are set to 4096. While, the number of outputs in fc8 is set according to the number of classes in each of the classification task. Filters size of 11×11 is used to yield conv1 with stride of 4. Filter size of 3×3 is used for other convolution layers with stride of 1, except for conv2 that uses filter size of 5×5. Local Response Normalization (LRN) is used after conv1 and conv2, similar in [28]. Following, pooling layers with size of 3×3 and stride of 2 after each LRN are conducted and conv5 for downsampling. Dropout is implemented after fc6 and fc7 for regularization as most of the hyper-parameters are concentrated in these layers. Rectified Linear Unit (ReLU) is used as the activation function for all weight layers, except for fc8 that uses softmax regression as activation and act as a multi-class classifier, to predict the paintings classification.

A. Training Details

Our models are trained using the stochastic gradient descent (SGD) with a batch size of 128 examples. The rest of the settings for SGD are momentum of 0.9 and weight decay of 0.0005. The update rule for weight w with respect to batch i was

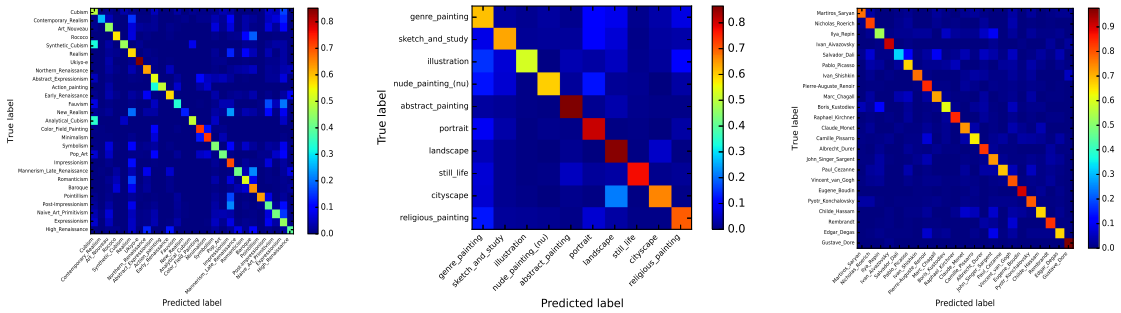
$$v_{i+1} = 0.9 \cdot v_i - \epsilon \cdot (\left\langle \frac{\partial L}{\partial w} \right\rangle_{B_i} + 0.0005 \cdot w_i) \quad (1)$$

$$w_{i+1} = w_i + v_{i+1} \quad (2)$$

where v is the velocity, ϵ is the learning rate, and $\left\langle \frac{\partial L}{\partial w} \right\rangle_{B_i}$ is the average over ith batch, B_i of the derivative of the objective function L with respect to w. We used multinomial logistic loss of softmax as our objective function.

For non-fine-tuning layers, we initialized the weights from zero-mean Gaussian distribution with a standard deviation of 0.01. Biases are initialized to 1, except for the first and third layers are set to zero. This is because setting biases to non-zero in some layers provide ReLUs with positive inputs to improve training [28].

Learning rate was initialized to 0.01 for all weights and 0.02 for bias, but during the fine-tuning, learning rate is reduced by a factor of 10 for all fine-tuning layers to avoid tampering the already well-learnt weights and biases. We reduced the learning rate for all layers by factor of 10, for every 5000 iterations prior to termination. We trained the model for 20,000 iterations for all experiments. We found that further training does not improve the results. The experiments were carried out using Caffe [29] on NVIDIA GTX 980 4GB GPU.



(a) Style (b) Genre (c) Artist

Fig. 1: Confusion matrix of the style, genre and artist classification task using the CNN-finetune model. The colorbar shows the normalized intensity. Best viewed in color.

1) Data Augmentation: We trained our network on the centered raw RGB values by subtracting the mean activity over the Wikiart dataset for each pixel. We used image translations and horizontal reflections as data augmentation during the training to reduce overfitting. Image translation is done by randomly cropping 227×227 patches from the 256×256 images. Each iteration will only crop **one** random patch for each image and the patches are then randomly mirrored. Note that different patches are cropped in each iteration. During validation, we extracted the centered cropped patches for testing without horizontal reflection.

2) Pre-training: In the fine-tuning process, we pre-trained the network using ImageNet dataset. In [21], they used the last layer of a pre-trained CNN with 1000 dimensional real-valued vectors as features. The extracted features are then compressed using Principle Component Analysis (PCA) and Support Vector Machine (SVM) is trained on top of them. Under similar setup, we conducted another set of experiment by stacking a new softmax layer on top of the pre-trained CNN without removing its last layer. In this sense, this network will be using the semantic-level features for the classification tasks, as oppose to other experiments (mid-level features).

IV. Experimental Results and Discussions

The results are summarized in Table I. CNN refers to an end-to-end deep model that we trained from scratch for each of the classification tasks. While CNN-nofine, CNN-SVM, CNN-1000, and CNN-finetune are pre-trained model based on the ImageNet dataset. CNN-nofine is a CNN model without the fine-tuning process, while CNN-finetune is a model that has been fine-tuned. CNN-SVM replaces the last layer with a SVM classifier instead of softmax layer. CNN-1000 preserves the final layer (the 1000-way multinomial output) of the pre-trained CNN.

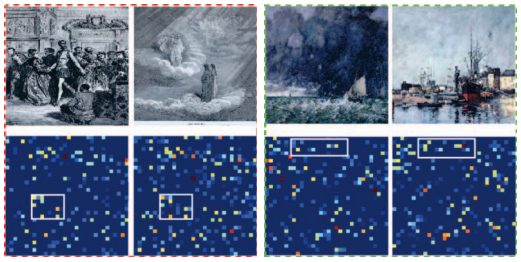
The results show that transfer learning helps in improving the training and the fine-tuning further improves the performance of the deep model to new tasks. Our CNN-finetune (68%) significantly outperform the state-of-the-art results (56%), and the accuracy between softmax and SVM classifiers are similar. However, it seems that preserving the final layer of 1000 dimensional vector does not result in a better performance. Also, we investigate the effect of network pruning (i.e compressing the CNN) to the system performance. For instance, we remove the fc7 layer (i.e CNN-fc6) and the results only deteriorate $\sim 2\%$ given that the pruning reduced the number of parameters from 61millions to 44millions. We also conduct another variant to the original Alexnet, changing the fc7 from 4096 to 1024 (i.e CNN-1024), and the results only differ by 0.5%. The main insight here is that, it seems that a better pruning strategy might able to compress the network further without affecting the system accuracy, as proven in a recent study by Han et al [30]. In order to further assess the performance of the CNN, we analyze the confusion matrix using the results from the CNN-finetune model; as well as visualizing the CNN features by the neurons' responses to the paintings in the next subsections.

Model	Accuracy (%)				Size
	Style	Genre	Artist	Overall	
CNN	42.96	65.45	54.39	54.27	61M
CNN-nofine	45.95	69.24	67.02	60.74	61M
CNN-SVM	44.17	69.18	67.17	60.17	61M
CNN-1000	43.56	68.38	64.55	58.83	61M
CNN-finetune	**54.50**	**74.14**	**76.11**	**68.25**	61M
CNN-fc6	51.51	72.11	74.26	65.96	44M
CNN-1024	53.38	73.75	76.02	67.72	48M
CNN-PCA-SVM [21]	21.99	49.98	33.62	35.20	-
Saleh and Elgammal [21]	45.97	60.28	63.06	56.44	-

TABLE I: Comparison of results on Wikiarts dataset for styles, genres, artists classification.

A. Confusion Matrix

Figure 1 shows the confusion matrix for each classification task. Among all, there are a few observations that worth attention. Firstly, in the *style* classification, it shows that the CNN can distinctly differentiate Ukiyo-e (85%) from the others. Ukiyo-e is a type of art that flourished in Japan that is very distinctive from other styles. However, the CNN is very poor in differentiating synthetic cubism (46%) and analytical cubism (50%), as both these styles are from the same root. This is similar to Rococo (56%) and Baroque (64%), as these two styles are historically related. This is also explained why



(a) Gustave Dore (b) Eugene Boudin

Fig. 2: The first row shows the artist's paintings and the second row shows the corresponding features in the feature space, where paintings from similar artists are close with each other.

the *style* classification has the poorest performance among the three classification tasks. Secondly, in the *genre* classification, it is not surprised that the top performers include portrait (81%) and landscape (86%) because CNN has been successful in face detection [31] and scene recognition [15].

Thirdly, it is interesting to see that the CNN perform the best in *artist* classification task. Is that mean that the CNN has or able to capture the "imagination" in paintings? To uncover the factors behind this, we investigate the best and worst performance. We discovered that the artists that CNN can recognize with high precision usually prefer certain techniques or objects in their paintings. For example, Gustave Dore uses mostly engraving, etching, and lithography, which results in greyish paintings as shown in Figure 2a. Eugene Boudin has many paintings that depict outdoor scenes, and most of his paintings was rendering marine and seashore (Figure 2b).

Meanwhile, Salvator Dalí a prominent Spanish best known for the striking and bizarre images, the CNN fails miserably (33%) and confuses his works to the greatest and most influential artists of the 20th century, Pablo Picasso. The most interesting part of this finding is, we found out that historically, Salvator Dalí made a number of works heavily influenced by Picasso, which is not known to us before this result. From our further investigations, a recent exhibit at the Salvador Dalí Museum in Florida (Feb. 2015) examined how these two artists influenced each other and Dr. William Jeffett, a Chief Curator of Exhibitions of the museum said *"The paintings look good together. The pieces really complement each other, which I think says a lot about the artists and their works, for example Picasso's Portrait of Olga, or Dalí's Portrait of My Sister. This wasn't just a contextual show where we were looking for academic or documentary links. There's a visual component here evident in the art, which supports what we're trying to say"*. So is this a hint that CNN is able to semantically link artists together? We leave this for future investigation.

B. Visualizing Neurons' Responses

Figure 3 visualizes the neurons' responses in the *genre* classification task. The visualization is done by averaging the neurons' value over the feature maps. As shown in the figure, in layer 1, neurons learned to recognize simple edge/blob

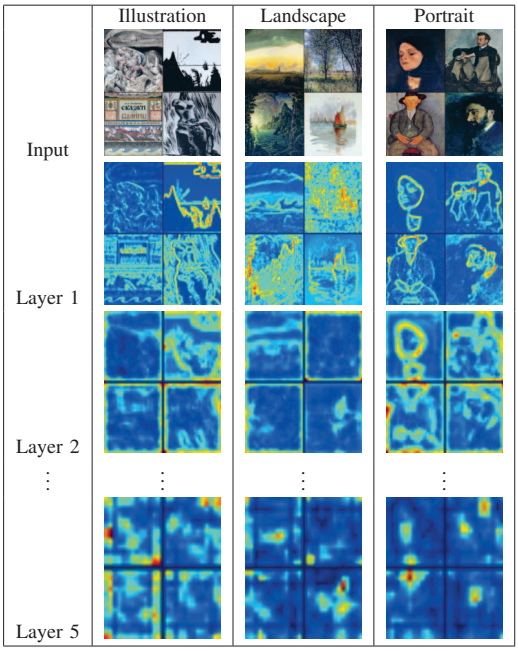


Fig. 3: Visualization of neurons' response in the *genre* classification task. Input neurons represent raw pixel values which are combined to edges in the lower layers (Layer 1). In the middle layers contours of parts thereof are built, which are finally combined to abstract features such as faces (Layer 5).

(low level features). As the layer goes higher, the neurons are learned to recognize texture pattern to complex object parts, such as face in Portrait. According to our observation, training a CNN for paintings classification tasks is tougher as paintings from the same group does not necessary have similar low to high level features (e.g Illustration). This also indicates why Illustration performs poorly in the *genre* classification task. For paintings that are more structured, the visualizations also show that CNN tends to find key objects or shapes for cues.

V. Conclusion and Future Work

In this work, we presented a study using CNN for fine-art paintings classification. We found that fine-tuning an ImageNet pre-trained CNN yields the best result and outperforms the state-of-the-art results. The confusion matrix shows that one of the reasons of misclassification was due to similar properties between the styles. Another reason that can be seen in the visualization is the difficulty in learning CNN model from paintings as the features extracted between lower and higher levels can be very different among the same group. In addition, we also found out that the CNN somehow could relate different artists together based their painting's style. For future work, we are interested in designing a better CNN model for paintings classification, and the possibly of semantically relate them together. Furthermore, we will investigate different visualization techniques for better understanding of how CNN extracts features from paintings.

REFERENCES

[1] G. Polatkan, S. Jafarpour, A. Brasoveanu, S. Hughes, and I. Daubechies, "Detection of forgery in paintings using supervised learning," in *IEEE International Conference on Image Processing ICIP*, 2009, pp. 2921–2924.

[2] E. J. Crowley and A. Zisserman, "In search of art," in *Computer Vision-ECCV 2014 Workshops*. Springer, 2014, pp. 54–70.

[3] E. Crowley and A. Zisserman, "The state of the art: Object retrieval in paintings using discriminative regions." in *BMVC*, 2014.

[4] T. Mensink and J. Van Gemert, "The rijksmuseum challenge: Museum-centered visual recognition," in *Proceedings of International Conference on Multimedia Retrieval*. ACM, 2014, p. 451.

[5] L. A. Gatys, A. S. Ecker, and M. Bethge, "A neural algorithm of artistic style," *arXiv preprint arXiv:1508.06576*, 2015.

[6] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems (NIPS)*, 2015, pp. 91–99.

[7] S. H. Lee, C. S. Chan, P. Wilkin, and P. Remagnino, "Deep-plant: Plant identification with convolutional neural networks," in *IEEE International Conference on Image Processing ICIP*, 2015, pp. 452–456.

[8] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," in *Computer Vision–ECCV 2014*. Springer, 2014, pp. 346–361.

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

[10] W. L. Hoo and C. S. Chan, "Zero-shot object recognition system based on topic model," *IEEE T. Human-Machine Systems*, vol. 45, no. 4, pp. 518–525, 2015.

[11] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," *arXiv preprint arXiv:1409.4842*, 2014.

[12] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International Journal of computer vision*, vol. 88, no. 2, pp. 303–338, 2010.

[13] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," 2009.

[14] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.

[15] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva, "Learning deep features for scene recognition using places database," in *Advances in neural information processing systems (NIPS)*, 2014, pp. 487–495.

[16] F. S. Khan, S. Beigpour, J. van de Weijer, and M. Felsberg, "Painting-91: a large scale database for computational painting categorization," *Machine vision and applications*, vol. 25, no. 6, pp. 1385–1397, 2014.

[17] C. R. Johnson Jr, E. Hendriks, I. J. Berezhnoy, E. Brevdo, S. M. Hughes, I. Daubechies, J. Li, E. Postma, and J. Z. Wang, "Image processing for artist identification," *IEEE Signal Processing Magazine*, vol. 25, no. 4, pp. 37–48, 2008.

[18] L. Shamir, T. Macura, N. Orlov, D. M. Eckley, and I. G. Goldberg, "Impressionism, expressionism, surrealism: Automated recognition of painters and schools of art," *ACM Transactions on Applied Perception (TAP)*, vol. 7, no. 2, p. 8, 2010.

[19] L. Shamir and J. A. Tarakhovsky, "Computer analysis of art," *Journal on Computing and Cultural Heritage (JOCCH)*, vol. 5, no. 2, p. 7, 2012.

[20] J. Zujovic, L. Gandy, S. Friedman, B. Pardo, and T. N. Pappas, "Classifying paintings by artistic genre: An analysis of features & classifiers," in *IEEE International Workshop on Multimedia Signal Processing*, 2009, pp. 1–5.

[21] B. Saleh and A. Elgammal, "Large-scale classification of fine-art paintings: Learning the right metric on the right feature," *arXiv preprint arXiv:1505.00855*, 2015.

[22] S. Karayev, M. Trentacoste, H. Han, A. Agarwala, T. Darrell, A. Hertzmann, and H. Winnemoeller, "Recognizing image style," *arXiv preprint arXiv:1311.3715*, 2013.

[23] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Computer Vision–ECCV 2014*. Springer, 2014, pp. 818–833.

[24] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, "Decaf: A deep convolutional activation feature for generic visual recognition," *arXiv preprint arXiv:1310.1531*, 2013.

[25] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Advances in Neural Information Processing Systems (NIPS)*, 2014, pp. 3320–3328.

[26] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Learning and transferring mid-level image representations using convolutional neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 1717–1724.

[27] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "Cnn features off-the-shelf: an astounding baseline for recognition," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2014, pp. 512–519.

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

[29] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," *arXiv preprint arXiv:1408.5093*, 2014.

[30] S. Han, J. Pool, J. Tran, and W. Dally, "Learning both weights and connections for efficient neural network," in *Advances in neural information processing systems (NIPS)*, 2015, pp. 1135–1143.

[31] S. Yang, P. Luo, C. C. Loy, and X. Tang, "From facial parts responses to face detection: A deep learning approach," in *International Conference on Computer Vision (ICCV)*, 2015.