




New Challenges of Face Detection in Paintings based on Deep Learning

Siwar Bengamra¹^a, Olfa Mzoughi²^b, André Bigand³^c
and Ezzeddine Zagrouba¹

¹ *LIMTIC, University of Tunis El Manar, Ariana, Tunisia*

² *Prince Sattam Bin Abdulaziz University, U.A.E.*

³ *LISIC, ULCO, Calais Cedex, France*

Keywords: Face Detection, Artworks, Tenebrism Style, Deep Learning, Convolutional Neural Network.


Abstract: In this work, we address the problem of face detection from painting images in Tenebrism style, a particular painting style that is characterized by the use of extreme contrast between light and dark. We use Convolutional Neural Networks (CNNs) to tackle this task. In this article, we show that face detection in paintings presents additional challenges as compared to classic face detection from natural images. For this, we present a performance analysis of three CNN architectures, namely, VGG16, ResNet50 and ResNet101, as backbone networks of one of the most popular CNN based object detector, Faster RCNN, to boost-up the face detection performance. This paper describes a collection and annotation of benchmark dataset of Tenebrism paintings. In order to reduce the impact of dataset bias, we propose to evaluate the effect of several data augmentation techniques used to increase variability. Experimental results reveal a detection average precision of 44.19% with ResNet101, while better performances have been achieved 79.48% and 83.94% with VGG16 and ResNet50, respectively.


1 INTRODUCTION


The evolution of computer vision-based study of visual art has been an extremely active research area over the last decades. Recently, the task of inferring properties of illumination distribution from art paintings has become important to discover and understand the history of art. In Tenebrism style, art historians were especially interested in estimating the illuminant position within a painting, and thereby in answering technical questions. For example, it is used to verify if there is a single source of light or multiple ones during painting. Is the source of light the one depicted or outside the picture? Is the painting executed under different studio conditions or even by possibly different artists? To answer these questions and many others technical ones, Stork and Johnson (Stork and Johnson, 2006) have focused on this issue and they have shown that the illuminant location estimation is mainly linked to the face viewpoint of people depicted in the painting.

Face detection is a fundamental and important task in a variety of computer vision applications such as person re-identification, surveillance system, facial expression recognition and facial image enhancement. Although it has been extensively studied over the past years, automatic face detection remains an important area of research due to the increasing need for accuracy improvements, especially in unconstrained environments, or in the presence of new domain applications. For that, we firstly focus on automatic face detection in Tenebrism paintings. However, this task represents different challenges as given in (Mzoughi et al., 2018). Tenebrism paintings are characterized by violent composition between light and dark. They also exhibit large variation in viewpoint, pose and occlusion. Compared to real scene photographs, there is a significant difference in appearance and dress of painted characters. Finally, the number of available painting images is limited.

Recently some authors tried to quantify painting styles as Van Gogh and Pollock ones ((L.Shamir, 2012)) with success. It appears that this task is particularly difficult using classic image processing attributes (in this paper some 42 to 80 features are used to correctly recognize styles), and the author shows

^a  <https://orcid.org/0000-0001-5546-5292>

^b  <https://orcid.org/0000-0001-8758-9740>

^c  <https://orcid.org/0000-0002-3165-5363>

the complexity of identification of aesthetic painting properties. Thus some authors investigate (machine learning (M.Fiorucci et al., 2020)) CNNs to avoid this problem (Qiao et al., 2019). They succeed in transferring ancient paintings to natural image and thus propose a new solution for painting processing.

With the great breakthrough of deep convolutional neural networks (CNNs) and the availability of large people-focused datasets, a new generation of more effective face detectors emerged to improve state-of-the-art performances and these CNN-based face detectors can be roughly divided in three categories: sliding window, two-stage and single stage detectors. The sliding window approach consists of scanning the image with rectangular window on multiple scales and applying a CNN-classifier to each sub-image (Sermanet et al., 2014). The sliding window process is simple but extremely slow. The two-stage detectors have similar design as Faster RCNN (Ren et al., 2015) by generating region of interest (ROIs) to filter out most of the background at first and then classifying each ROI, as well as further regressing them to the ground-truth locations. In this scope, we report recent works: FA-RPN (Najibi et al., 2018) and DSFD (Wu et al., 2019). In (Najibi et al., 2018) authors proposed enhancing the robustness of face detection via a novel strategy for generating region proposals. On the other hand, the one-stage detectors directly output bounding boxes and confidences without region proposal parts and include YOLO (Redmon et al., 2016), SFDet (Zhang et al., 2019a), PyramidBox (Tang et al., 2018), DSFD (Li et al., 2018), S3FD (Zhang et al., 2017) and SSH (Najibi et al., 2017). While one-stage methods made detection at a real time speed, two-stage methods have been proved to be more accurate than other methods (Li et al., 2019; Zhang et al., 2019b; Quang and Fujihara, 2019). Because the real-time speed is not so important for the purpose of our domain application, we use Faster RCNN (Ren et al., 2015), the top performing two-stage detectors in recent years.

Comparing to classic approaches, deep learning-based methods successfully inherit powerful feature extraction abilities thanks to different existing CNN model architecture. Thus, the detection performance depends significantly on the backbone model used for feature extraction. Thereby, the main contributions of this work consist in: (1) a collection of new challenging Tenebrism dataset to advance current research in Artwork and computer vision, (2) a comparative study of fine-tuned popular CNN models to assess the best model for face detection in Tenebrism context and (3) investigation of data augmentations to evaluate their influence on face detection.

The rest of the paper is organized as follows. Section 2 briefly presents and discusses common deep convolutional neural network architectures. For face detection in painting, we adopt the deep learning framework Faster RCNN, which we describe in section 3. The description of the new dataset and the experimental results are provided in sections 4 and 5, respectively. Finally, we draw conclusions and recommend future directions in section 6.

2 POPULAR DEEP CNN ARCHITECTURES

2.1 VGGNet based Architecture

VGGNet was introduced by the Visual Geometry Group (VGG) of the university of Oxford in 2014 (Simonyan and Zisserman, 2015). It achieves the first place on image localization task and the second place on the classification task at 2014-ImageNet Large Scale Visual Recognition Challenge (ILSVRC) competition. The VGG architecture contains subsequent convolutional layers, each of them uses the ReLU activation function, followed by max-pooling layers and three fully connected layers. Max-pooling layers are applied at different steps in the architecture and are used to reduce the size of the volume. The final layer is a Softmax layer used for classification. VGG net have realized an improvement with regard AlexNet by utilizing receptive field (i.e. kernel-sized filters of 3×3) much smaller than that of AlexNet (Krizhevsky et al., 2012) (11×11) in order to provide better feature extraction. There are two versions of this architecture according to the number of convolutional layers: VGG16 and VGG19. The main limitation of VGG networks is that they are very large models in terms of the number of trainable parameters (138 millions). Hence VGG nets require extensive computational and memory resources which make it slow to train.

2.2 ResNet based Architecture

A ResNet, is a deep Convolutional Neural Network with residual learning elements, and was introduced by (He et al., 2016) and won prize in the ImageNet Large Scale Visual Recognition Challenge 2015 (Russakovsky et al., 2015) and Microsoft Common Objects in Context 2015 (Lin et al., 2014). Unlike traditional CNNs where the output of the convolution layer is the input for the next convolution layer, ResNet uses residual learning with identity as shortcut connections to skip training of few layers in the forward

feeding on an input. In fact, shortcut connection consists in adding the activation from a previous layer as a residue to the activation of a deeper layer in the network in order to predict the desired output. The popular ResNet50 is a 50 layer residual network with 49 convolutional layers and one fully connected layer at the end. There are other variants of ResNet according to the different numbers of layers: 34, 101, 152.

2.3 Discussion

We remark that most deep CNN architectures typically follow the simplest type of model, the sequential model, as a linear stack of layers (e.g. LeNet (LeCun et al., 1995; El-Sawy et al., 2016), AlexNet (Krizhevsky et al., 2012) and VGGNet (Simonyan and Zisserman, 2015)). The two concepts, local response normalization (LRN) and dropout, are introduced with AlexNet to improve the generalization by reducing overfitting. The ReLU activation function was also employed to improve the convergence performance instead of the conventional activation functions like "tanh" and "sigmoid" functions, by alleviating the problem of vanishing gradients for positive values (Filonenko et al., 2017; Khan et al., 2019). The key innovations in the VGG architecture were the reduce of filter size and the increase of depth (Khan et al., 2019). In fact, the small size of convolutional filters involves the use of more ReLU units that makes the decision function more discriminative. Although, increasing the depth with VGG leads the network to better performances by extracting rich and diverse features, it makes training more difficult and computationally expensive requiring supercomputing infrastructure for producing results (Khan et al., 2019). Recently, ResNet architecture (He et al., 2016) has shown to be effective for dealing with the problem of vanishing gradients by using skip connections which facilitates larger gradient flow to initial layers during backpropagation (Varshaneya et al., 2019; Wan et al., 2018).

We thus can summarize the importance to understand how CNN architecture design influences model accuracy. Many other factors, such as data augmentation, training dataset, Intersection-Over-Union threshold, loss function and hyperparameters (e.g. batch size and learning rate), can also impact the detection performance

3 FASTER RCNN

Faster-RCNN (Ren et al., 2015) is one of the most well-known object detection algorithms which is

composed of a *backbone* network and two *subnetworks*. The backbone is responsible for extracting convolutional features. The first subnetwork, called Region Proposal Network (RPN), is devoted to propose candidate object bounding boxes; the second subnetwork, which is in essence Fast R-CNN (Girshick, 2015), associates features to each generated candidate box to perform classification and bounding-box regression.

3.1 Feature Network

The backbone network is used to extract the 2D feature map over the entire input image. This network consists of 2D convolutional layers and max pooling layers obtained from a base convolutional network such as VGG16 minus a few last layers.

3.2 Region Proposal Network

The Region Proposal Network (RPN) is then applied to get N proposals called the Region of Interests (ROIs) that are likely to contain any object from the backbone's output: the convolutional 2D feature map. For each sliding window (e.g. pixel location) over the input feature map, the RPN first generates nine anchors with different size scales (128, 256, 512) and three aspect ratios (1:1, 1:2, 2:1). For each anchor, we assign a ground-truth label to 1 if it has an Intersection-over-Union (IOU) score greater than 0.5 with one ground-truth box, and a label equals to 0 otherwise negative. Every anchor is then mapped to a low-dimensional feature vector that will be fed into two competitive fully-connected layers - a box-classification layer and a box-regression layer. The classification layer is responsible for checking if the anchor (bounding box candidate) belongs to object class (= positive) or not (=negative). The IOU is calculated by Eq. 1

$$IOU = \frac{A_{proposal} \cap A_{ground-truth}}{A_{proposal} \cup A_{ground-truth}} \quad (1)$$

where $A_{proposal}$ and $A_{ground-truth}$ are the area of the proposals (i.e. anchors) and ground-truth bounding boxes, respectively. The box regression layer tries to adjust the boundaries of the proposals according to ground-truth. The outputs will be region proposals coordinates representing two diagonal corners for each proposal (top-left and bottom-right) and probabilities representing how likely the region proposal is to be an object (i.e. objectness scores). Finally, at the ROI proposal layer, the regions of interests (ROIs) are gathered in the descending order of objectness

score. Non-maximum suppression is used to combine regressed anchors before selecting ROIs from anchors to avoid duplicated ROIs. To optimize the RPN performance (adjust the weights in the RPN) during training, both classification loss and bounding box regression loss are defined and given by the equation 2.

$$L(p_i, t_i) = \frac{1}{N_{cls}} \sum_i L_{cls}(p_i, p_i^*) + \lambda \frac{1}{N_{reg}} \sum_i p_i^* L_{reg}(t_i, t_i^*) \quad (2)$$

where i is the index of the anchor in the mini-batch and λ is a balancing constant. The first term in 2 is the classification loss over two classes and is defined by 3. p_i is the probability from the classification branch for anchor i and p_i^* is the assigned ground-truth label.

$$L_{cls}(p_i, p_i^*) = -p_i^* \log(p_i) - (1 - p_i^*) \log(1 - p_i) \quad (3)$$

The second term is the regression loss of bounding box activated only if the anchor contains an object (i.e. $p_i^* = 1$). The definition for this regression loss is described by equation 4:

$$L_{reg}(t_i, t_i^*) = \sum_i \text{smooth}_{L_1}(t_i - t_i^*) \quad (4)$$

The variables t_i and t_i^* represent the four coordinates of predicted bounding box and the ground-truth coordinates, respectively .

3.3 Detection Network

In this stage, each ROI is classified and its bounding box is refined (i.e. regressed) using the Fast RCNN network (Girshick, 2015). At first, ROI Pooling layer is used to normalize the candidate regions (proposals). The obtained feature-map regions will then be flattening into a fixed-length feature vector regardless of input feature map and proposal sizes. Finally, the feature vectors are put into a sequence of fully connected layers, which includes a softmax layer and a linear regression layer, to conduct classification and regression.

4 EXPERIMENTAL SETUP

For experimental setup, we used Google Colab Nvidia Tesla K80 GPU for training. The tests were performed using a workstation powered by an Intel core i7- (3.9 GHz) processor, with 32 GB RAM, and an NVIDIA GeForce GTX 1650 GPU with a graphics memory of 4GB. We adopt a keras implementation of Faster RCNN (Lufan, 2019) using TensorFlow library to train the deep learning models and predict face bounds and objectness score.

The performance of our system is evaluated first of all in terms of the Intersection-over-Union (IOU, Eq. 1). So if IOU outperforms a threshold value, the face proposal is considered as true positive, or if not as a false positive. TP (True Positive) indicates the number of correct faces detected, FP (False Positive) indicates the number of wrong face detected, TN (True Negative) indicates the ground-truth faces not detected, and FN (False Negative) indicates the number of all possible faces that were correctly not detected. To give an overall insight of the performance of the face detector, we measured the following commonly used metrics: the precision/recall curve and the mean average precision (mAP) as computed in the Pascal VOC challenge (Everingham et al., 2009).

4.1 Tenebrism Dataset

Face detection in paintings in general, and in Tenebrism style in particular is a new topic of research. There is no benchmark dataset for this problem. For that, we collect our dataset from two main sources:

- 304 images are sourced principally from Google images, WikiArt.org and Pinterest,
- The other 105 images are collected from Github (Meier, 2018) and used in the problem of hand detection.

The dataset contains 409 color painting images belonging to the Tenebrism style. They contain in total 1159 faces and each image holds at least one face. The dataset may contain the same painting but with variable capture conditions (see Figure 1).

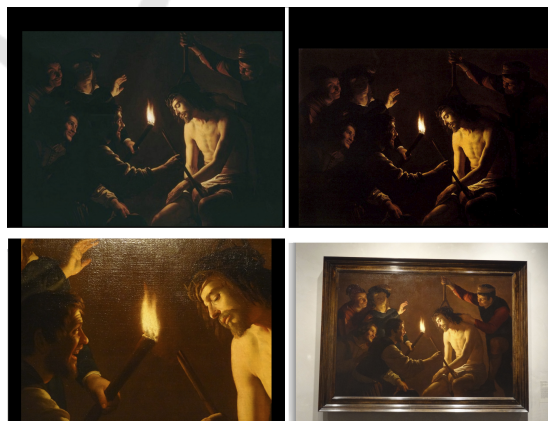


Figure 1: The same painting taken with variable capture conditions.

Face detection in this dataset is challenging for different reasons (Figure 2). First, it contains different face views: frontal, mid-profile and profile (right and left). In addition to that, there are various faces partially and heavily occluded with hair, clothes or



Figure 2: Challenges in face annotation: low resolution (a, green), occlusion (b, blue) and difficult poses (b, c, red).

persons. Another aspect which is particular in this dataset is that it holds a significant difference in appearance and dress of painted characters as well as in the scene in general, compared to real-scene photographs. Moreover, it is characterized by violent contrasts of light and dark. Finally, the dataset contains several low-resolution face images. In such conditions, we aim to study if pre-trained photograph-based face detectors could realize successful results in such images. In fact, a good face detector should detect a face whatever it is photographed or painted. The dataset is annotated using the graphic image annotation tool labelling available at (Tzutalin., 2015). The annotation process consists of drawing bounding boxes around faces in the images, then generating automatically XML files to store location details of the faces in the images.

4.2 Training Settings

We follow the original Faster RCNN to set the hyperparameters for end-to-end training. The weights of the used deep learning models were initialized from models that are pre-trained on the natural image dataset ImageNet (Deng, 2009). To minimize overfitting during training and to improve generalization to unseen paintings, several data augmentation techniques were applied randomly: contrast changes (C), horizontal flipping (HF), vertical flipping (VF) and rotation (R). Augmentation is performed on-the-fly for each batch since it can generate more unique training images than offline augmentation, which can improve generalization capability (O’Gara and McGuinness, 2019). The goal of contrast augmentation is to apply random changes to contrast for improving the robustness of CNN models whilst preserving geometry.

For that, we used contrast limited adapted histogram equalization (CLAHE) (Zuiderveld, 1994). Training images with the associated bounding boxes around faces are rotated with angle in $[-45^\circ, 45^\circ]$.

5 RESULTS AND DISCUSSION

In this section, first, we show limitation of a photograph-based face detector. Then, we compare performance of three different architectures fine-tuned on our Tenebrism Dataset. In fact, we use ResNet and VGGNet since most detection networks utilize them as the basic feature extraction module at present (Zhang et al., 2020; Chi et al., 2019). So, VGG16 was selected first, as used in the original paper (Ren et al., 2015), and two ResNet configurations (ResNet50 and ResNet101) were investigated for the new task. Face detection is then evaluated by k-fold cross-validation process. Finally, we evaluate the effects of data augmentations on the face detection performances.

5.1 Evaluation of Transfer Learning

A great success has been realized using faster-RCNN face detector trained on photographs. It has realized a mAP of 97.79% (Wu et al., 2019). We expect that a good face detector should recognize faces regardless they are photographed or painted. To investigate this issue, the following experiment has been established. We train Faster RCNN on the AFLW dataset (which is a famous dataset of photograph faces). Then we test this model, called Model 1 in the sequel, on the Tenebrism dataset. We obtain a mean average precision (AP) score of 29.05% (see table 1) which

proves the limitation of photograph-based face detector in the context of Tenebrism paintings. We can conclude that these images exhibit some specific features that makes them different from photographs. For that, we investigate the use of transfer learning in two levels. In the first level we apply transfer learning from Model 1 by retraining only the last classification stage (i.e. the fully connected layers) on our specific Tenebrism dataset. The second level consists of retraining all layers of model 1 on our target dataset. The two-level models are noted respectively Model 2 and 3. Results are shown in table 1. By employing transfer learning, model 2 achieves 19.3% improvements in mean average precision. In addition, when the model 1 is trained with a smaller number of painting images, high face detection performances are achieved (79.6 %), thereby illustrating the power of transfer learning to make models generalize well in the task of face detection in paintings, even with a limited number of training dataset. This experiment clearly demonstrates the specificity of painting images compared to natural images. Figure 3 depicts the Precision-Recall curves of the three deep learning models previously mentioned for face detection from Tenebrism paintings. We remark that model 1 performs poorly by detecting a large number of incorrect faces (low precision) and missing most ground truth faces (low recall). The model 2 has high precision but low recall, meaning that a significant number of ground truth faces are not detected. When training the model 1 on our target dataset, the resultant model 3 achieves a significant gain in recall.

Table 1: Evaluating the effect of Transfer Learning.

Experiment	mAP_{75}
Model 1	29.05 %
Model 2	48.35 %
Model 3	79.6 %

5.2 Effects of Feature Extractors

The proposed system implements the Faster RCNN meta-architecture with different feature extractors to deal with the face detection from Tenebrism images. Table 2 shows comparative face detection performances of fine-tuned backbone architectures on our Tenebrism dataset. First, we observe that the learned models ResNet50 and VGG16 achieved high performances when tested on Tenebrism images, where the ResNet101 model leads to inferior results. Note also that Faster RCNN with ResNet50 as feature extractor slightly exceeds the face detection average precision obtained with VGG16. This performance difference can be initially explained by the important

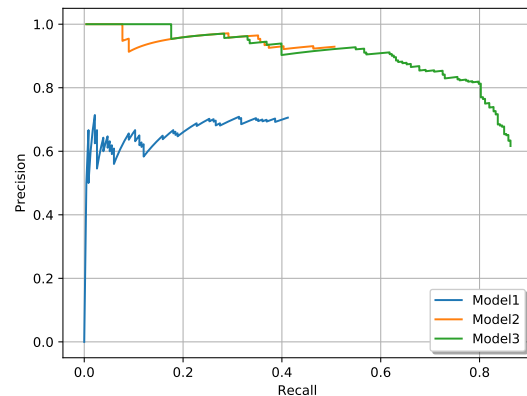


Figure 3: Performances of deep learning models with 0.75 IOU threshold.

role of shortcut connections in ResNet architecture which prevent loss of information transmitted in the layer. The increase of the network depth (i.e. number of layers) can also justify the high performance rate obtained with ResNet50 (83.94%) by learning more complex features. However, we remark that using deeper network like ResNet101, may require more epochs and certainly bigger training dataset for convergence. Based on these results, we believe that visualizing the internal features of Faster RCNN based ResNet50 as perceptible patterns can be helpful to understand the internal working mechanism and contribute to significant advances in face detection from Tenebrism paintings.

Table 2: Face detection performances for Faster RCNN with ResNet50, ResNet101 and VGG16 as backbones. mAP_{50} and mAP_{75} are for IOU threshold 0.5 and 0.75, respectively.

Backbone architecture	mAP_{50}	mAP_{75}
Resnet50	83.94%	74.64%
Resnet101	44.19%	31.45%
VGG16	79.48%	70.01%

Figure 4 shows some examples of face detection outputs obtained by Faster RCNN trained on our Tenebrism dataset. The detection results using the two best performing models ResNet50 and VGG16 are depicted by orange and purple bounding boxes, respectively, with confidence scores indicating the system's confidence on the face detection result. Ground-truth faces are represented by green bounding boxes. Faster RCNN is able to localize faces under different illumination conditions and for different viewpoints accurately. However, using VGG16, Faster RCNN fails to detect objects which can be localized with Faster RCNN based ResNet50 (see Figure 4). The learning process of ResNet50 and VGG16 with Tenebrism images can be analyzed through figure 5,



Figure 4: Qualitative results. (a) original images and outputs of Faster RCNN based (b) ResNet50 and (c) VGG16 with some false positives and false negatives.

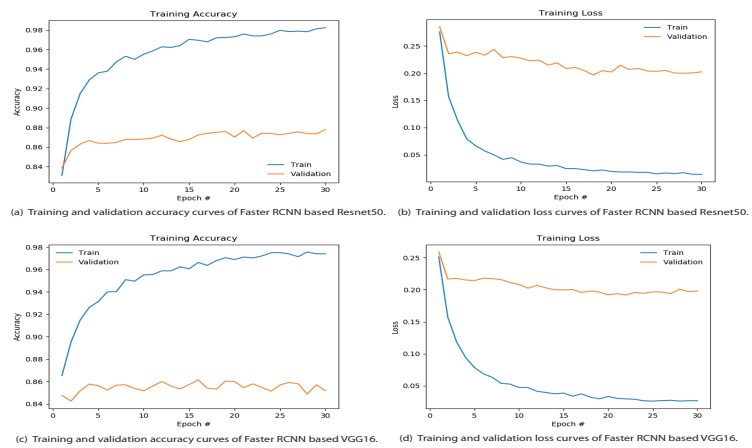


Figure 5: Training accuracy and loss of Faster RCNN based ResNet50 and VGG16.

Table 3: Five-fold cross validation diagram.

Data	Number of faces		Performances(mAP)
	Training set	Validation set	
Fold 1	261(752 faces)	66(175 faces)	73.89%
Fold 2	261(741 faces)	66(186 faces)	64.68%
Fold 3	262(738 faces)	65(189 faces)	68.67%
Fold 4	262(735 faces)	65(192 faces)	83.08%
Fold 5	262(742 faces)	65(185 faces)	67.89%
Average	262(742 faces)	65(185 faces)	71.64%

Table 4: Influence of data augmentations in face detection (Experiments with IOU > 0.5).

Data augmentations	VGG16	ResNet50
Without Data Augmentation	79.84%	83.94%
Contrast enhancement (CE)	80.13%	81.11%
Horizontal Flipping (HF)	80.85%	84.06%
Vertical Flipping(VF)	75.62%	83.99%
Rotation (R)	69.89%	86.51%

showing the accuracy/loss curves. It can be observed that the training and validation accuracy provided by ResNet50 are relatively high. We can also deduce that the validation error consistently decreases with the training error implying that no overfitting is observed.

5.3 Evaluation with Cross Validation

In order to evaluate the effectiveness of the model, we conduct k-fold cross-validation experiments with the commonly used $k = 5$. We performed experiments with Faster RCNN based ResNet50 since it produced the best results (section 5.2). In detail, the training dataset was divided into five equal parts (folds) randomly, and in each round, one of these five parts is used as evaluation set and the remaining four parts are used as training set. The operation diagram of the five-fold cross-validation with performance measures are illustrated in Table 3. As a result, we obtained an interesting mean average precision of $71.64\% \pm 10$ in five fold cross-validation that can approve the stability of the current face detection model.

5.4 Effects of Data Augmentation

To investigate the effects of data augmentation on the performances of the Faster RCNN network, we evaluate separately each image data augmentation technique mentioned in section 4.2. Table 4 shows that the use of online data augmentation can contribute to improve the performances compared to the previously trained models without data augmentation. For example, compared to the non-data-augmented Faster RCNN based ResNet50, the HF, VF and R improve

the detection results by 0.12%, 0.05% and 2.57%. Thus we obtain best results with ResNet50 and rotation (mean average precision of 86.51%). Unfortunately, we also observe that these random augmentations can degrade the face detection performances (from 83.94% to 81.11% with CE), which may be explained by a possibility of an intra-class imbalanced data created due to these naive augmentations. This experiments motivated us to focus on proposing more effective augmentation techniques according to the specific Tenebrism style of images.

6 CONCLUSION

In this work, we employed the deep learning framework Faster RCNN to detect faces from Tenebrism paintings. Firstly, we described the collection and annotation of a limited benchmark, namely Tenebrism dataset for existing methodologies comparison. Then, we show that fine-tuning Faster RCNN with different backbones, ResNet50 and VGG16, provide impressive results that can be helpful for further advances in face detection from Tenebrism paintings. Although, online data augmentation makes it possible to improve face detection, performances can be deteriorated. So in the future, we will continue to study and experiment Tenebrism specific data augmentations. We also plan to deploy the detection technique for other parts of body in the paintings towards further progress of art technique understanding by art historians. So far, the expected impact and outcomes of automatic human part detection from ancient art paintings should help art historians to better understand illumination techniques. A better comprehen-

sion of the specific features characterizing paintings is also expected to explain our results, since we do not have theoretical tools to explore that way at the moment. Thus a great number of application works can be explored in the future.

REFERENCES

- Chi, C., Zhang, S., Xing, J., Lei, Z., Li, S., and Zou, X. (2019). Selective refinement network for high performance face detection. *ArXiv*, abs/1809.02693.
- Deng, J. (2009). A large-scale hierarchical image database. In *CVPR 2009*.
- El-Sawy, A., El-Bakry, H. M., and Loey, M. (2016). Cnn for handwritten arabic digits recognition based on lenet-5. In *AI SI*.
- Everingham, M., Gool, L. V., Williams, C. K. I., Winn, J. M., and Zisserman, A. (2009). The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88:303–338.
- Filonenko, A., Kurnianggoro, L., and Jo, K.-H. (2017). Comparative study of modern convolutional neural networks for smoke detection on image data. *2017 10th International Conference on Human System Interactions (HSI)*, pages 64–68.
- Girshick, R. B. (2015). Fast r-cnn. *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1440–1448.
- He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778.
- Khan, A., Sohail, A., Zahoor, U., and Qureshi, A. S. (2019). A survey of the recent architectures of deep convolutional neural networks. *Artificial Intelligence Review*, pages 1–62.
- Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *NIPS*.
- LeCun, Y., Jackel, L. D., Bottou, L., Cortes, C., Denker, J. S., Drucker, H., Guyon, I., Muller, U. A., Sackinger, E., Simard, P. Y., and Vapnik, V. (1995). Learning algorithms for classification: A comparison on handwritten digit recognition. In Oh, J., Kwon, C., and Cho, S., editors, *Neural networks*, pages 261–276. World Scientific.
- Li, J., Wang, Y., Wang, C., Tai, Y., Qian, J., Yang, J., Wang, C., Li, J., and Huang, F. (2018). DSFD: dual shot face detector. *CoRR*, abs/1810.10220.
- Li, R.-Q., Bian, G.-B., Zhou, X.-H., Xie, X., Ni, Z.-L., and Hou, Z.-G. (2019). A two-stage framework for real-time guidewire endpoint localization. In *MICCAI*.
- Lin, T.-Y., Maire, M., Belongie, S. J., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *ECCV*.
- L. Shamir (2012). Computer analysis reveals similarities between the artistic styles of Van Gogh and Pollock. *Leonardo*, 45(2):149–154.
- Lufan, C. (2019). Keras implementation of faster r-cnn. <https://github.com/moyiliyi/keras-faster-rcnn>.
- Meier, G. J. (2018). Detecting hands in renaissance era paintings through a combination of multiple cues. Master's thesis, Utrecht University.
- M.Fiorucci, M.Khoroshiltseva, M.Pontil, A.Traviglia, Bue, A., and S.James (2020). Machine learning for cultural heritage: a survey. *Pattern Recognition Letters*, 133:102–108.
- Mzoughi, O., Bigand, A., and Renaud, C. (2018). Face detection in painting using deep convolutional neural networks. In *Advanced Concepts for Intelligent Vision Systems (ACIVS)*, pages 333–341, Cham. Springer International Publishing.
- Najibi, M., Samangouei, P., Chellappa, R., and Davis, L. S. (2017). SSH: single stage headless face detector. *CoRR*, abs/1708.03979.
- Najibi, M., Singh, B., and Davis, L. S. (2018). FA-RPN: floating region proposals for face detection. *CoRR*, abs/1812.05586.
- O'Gara, S. and McGuinness, K. (2019). Comparing data augmentation strategies for deep image classification. In *IMVIP 2019: Irish Machine Vision and Image Processing Conference Proceedings*.
- Qiao, T., Zhang, W., Zhang, M., Ma, Z., and Xu, D. (2019). Ancient painting to natural image: A new solution for painting processing. *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 521–530.
- Quang, N. V. and Fujihara, H. (2019). Revisiting a single-stage method for face detection. *2019 14th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2019)*, pages 1–8.
- Redmon, J., Divvala, S. K., Girshick, R. B., and Farhadi, A. (2016). You only look once: Unified, real-time object detection. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 779–788.
- Ren, S., He, K., Girshick, R. B., and Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39:1137–1149.
- Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M. S., Berg, A. C., and Li, F.-F. (2015). Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115:211–252.
- Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., and LeCun, Y. (2014). Overfeat: Integrated recognition, localization and detection using convolutional networks. *CoRR*, abs/1312.6229.
- Simonyan, K. and Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556.
- Stork, D. G. and Johnson, M. K. (2006). Computer vision, image analysis, and master art: Part 2. *IEEE Multi-Media*, 13(4):12–17.

- Tang, X., Du, D. K., He, Z., and Liu, J. (2018). Pyramidbox: A context-assisted single shot face detector. *CoRR*, abs/1803.07737.
- Tzutalin. (2015). Labelimg. <https://github.com/tzutalin/labelImg>.
- Varshaneya, V., Balasubramanian, S., and Gera, D. (2019). Res-se-net: Boosting performance of resnets by enhancing bridge-connections. *ArXiv*, abs/1902.06066.
- Wan, S., Liang, Y., and Zhang, Y. (2018). Deep convolutional neural networks for diabetic retinopathy detection by image classification. *Comput. Electr. Eng.*, 72:274–282.
- Wu, W., Yin, Y., Wang, X., and Xu, D. (2019). Face detection with different scales based on faster r-cnn. *IEEE Transactions on Cybernetics*, 49(11):4017–4028.
- Zhang, S., Chi, C., Lei, Z., and Li, S. (2020). Refineface: Refinement neural network for high performance face detection. *IEEE transactions on pattern analysis and machine intelligence*, PP.
- Zhang, S., Wen, L., Shi, H., Lei, Z., Lyu, S., and Li, S. Z. (2019a). Single-shot scale-aware network for real-time face detection. *Int. J. Comput. Vision*, 127(6–7):537–559.
- Zhang, S., Zhu, X., Lei, Z., Shi, H., Wang, X., and Li, S. Z. (2017). S³fd: Single shot scale-invariant face detector. *CoRR*, abs/1708.05237.
- Zhang, Y., Wang, J., Miao, Z., Li, Y., and Wang, J. (2019b). Shuffle single shot detector. In *ICIC*.
- Zuiderveld, K. J. (1994). Contrast limited adaptive histogram equalization. In *Graphics gems*.