EDGE-AWARE CONTEXT ENCODER FOR IMAGE INPAINTING

Liang Liao[1,2], *Ruimin Hu*[2,3], *Jing Xiao*[4], *Zhongyuan Wang*[3]

[1]State Key Laboratory of Software Engineering, Wuhan University, China
[2]National Engineering Research Center for Multimedia Software, Wuhan University, China
[3]Hubei Provincial Key Laboratory of Multimedia and Network Communication Engineering, China
[4]Collaborative Innovation Center of Geospatial Technology, China

ABSTRACT

We present Edge-aware Context Encoder (E-CE): an image inpainting model which takes scene structure and context into account. Unlike previous CE which predicts the missing regions using context from entire image, E-CE learns to recover the texture according to edge structures, attempting to avoid context blending across boundaries. In our approach, edges are extracted from the masked image, and completed by a full-convolutional network. The completed edge map together with the original masked image are then input into the modified CE network to predict the missing region. The experiments demonstrate that E-CE can generate images with better shapes and structures than CE.

Index Terms— image inpainting, edge extraction, convolutional network, complex scene

1. INTRODUCTION

As being firstly raised in [1] for art restoration, image inpainting is used to refer to the process of restoring missing or damaged areas in an image. Classical inpainting methods are often based on local or non-local information to recover images [2-5], which are commonly assumed to be insufficient when missing region is large. Recently, Context Encoder (CE) [6] has shown promise as prediction model of high-level context for completing images with large holes. A flurry of work has proposed improvements over original CE, such as adding pixel domain constraints [7,8], adding feature domain constraints [9-11], and adding post-processing to deal with high-resolution [12], image blending [13] and image restoration [14].

While these approaches can predict correct context and generate natural look missing region for some datasets with specific image contents, such as faces or streets, complex natural scenes are still challenging. For instance, the objects in these images tend to be deformed or mixed with the surrounding environment, and do not look realistic or recognizable (Fig. 1 (b)).

One fundamental limitation of these methods is that the network attempt to understand the context of the entire image, lacking the ability to handle the environmental complexity. Yet, natural images contain composite structures and textures, and the textures are usually regions with homogenous patterns bounded by structures. Uniformed processing them results in clutter structures and mixing textures.

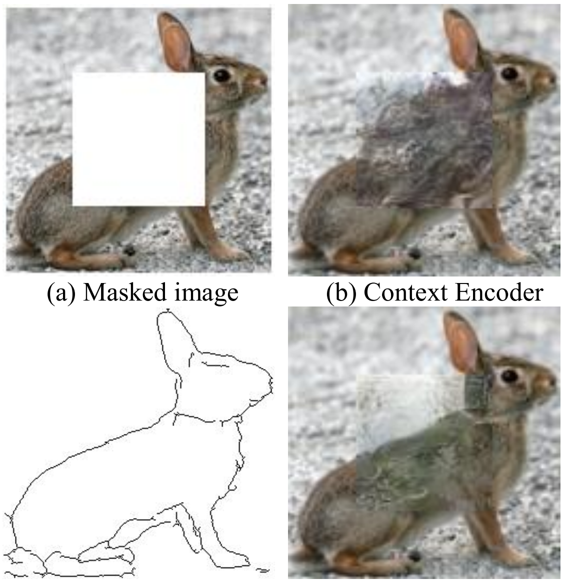
(a) Masked image (b) Context Encoder
(c) Binary edges (d) Conditional on edges

Fig. 1. Visual illustration of the task. (a) input masked image; (b) inpainting result from CE; (c) binary edges; (d) edge-aware inpainting result.

To handle composite textures and structures, we propose an Edge-aware Context Encoder (E-CE) model to predict missing region. Edge maps can keep the major structures while removing the weak correlation between different textures, thus can be used as a constraint in the texture prediction of missing region. During the inpainting process, we firstly develop a method to draw rough edges in the missing region based on the edges extracted from known area. Then the entire edge map is input together with the known region to predict the textures in the missing region. The proposed approach is naturally consistent with the order of drawing by a painter. Fig. 1 (c) and (d) represent the edges and edge-aware inpainting result of Fig. 1 (a) respectively.

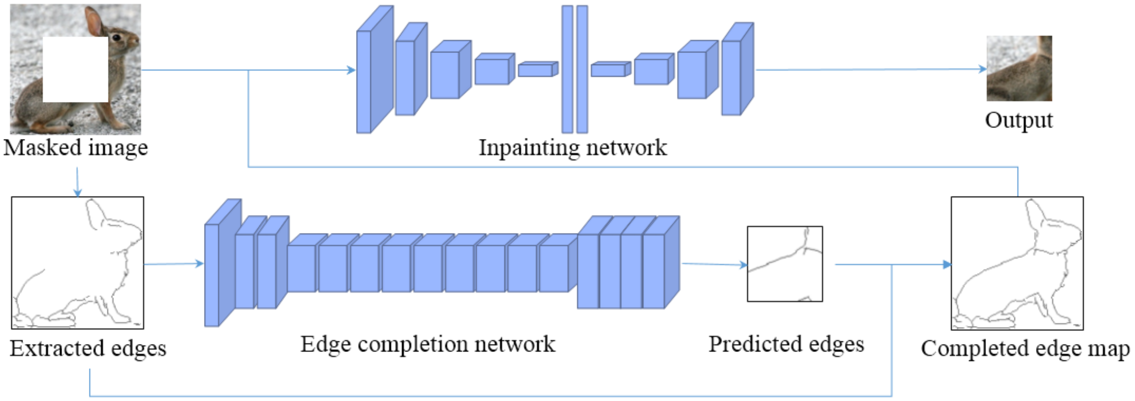

Fig. 2. Framework of edge-aware context encoder for image inpainting.

We evaluate our approach on two datasets: 137K Amazon Handbag images [15] and 100K-ImageNet [16]. The results show that E-CE is able to correctly predict structures and shape of objects in the missing region.

2. EDGE-AWARE IMAGE INPAINTING STRATEGY

In this section, we describe the proposed model for image inpainting. Fig. 2 shows the proposed framework that consists of an edge map generation part and a CE based inpainting part. Given a masked image, the edge map generation part extracts dominant edges from known region, following by an edge completion network to obtain a completed edge map. Then the edge map together with the masked image is input into a modified CE network to fill textures in the missing region. The edges provide guidance in the prediction of the textures.

2.1. Generation of edge image

Generation of edge image consists of two steps: edge extraction from masked image and completion of edges in the missing region.

2.1.1. Edge extraction

In order to provide boundary and main structures for the texture inpainting, we prefer to extract the outlines of objects or texture patches rather than fine texture details. Traditional Sobel or Canny [17] based method only considers the local changes of color, light or gradient. They cannot generate satisfactory result in complex environment as the case in our study. Therefore, we adopt a recently proposed CNN based Holistically-Nested Edge Detection (HED) model [18]. It deals with the holistic image in the process, enabling an extraction of high-level boundary information.

In our work, edges are firstly extracted using already trained HED model. Then we adopt standard non-maximum suppression and edge thinning for the post-processing.

2.1.2. Edge map completion

Considering that the main purpose of edge completion is to recover the connectivity between edges, we adopted a fully convolutional network [19]. An overview of the network architecture can be seen in Table 1. The input of the completion network is a single channel image with a mask, and the output is the predicted edges. The network architecture follows an encoder-decoder structure, and decreases the resolution using strided convolutions to reduce the memory usage and computational time.

Table 1. Architecture of the edge completion network. After each convolution layer and deconvolution layer, except the last one with a tanh function, there are a Batch Normalization layer and a Rectified Linear Unit layer. “Outputs” refers to the number of output channels for the output of the layer.

Layer	Type	Kernel	Stride	Padding	Outputs
1	conv.	5 × 5	1 × 1	1 × 1	64
2	conv.	3 × 3	2 × 2	1 × 1	128
3	conv.	3 × 3	1 × 1	1 × 1	128
4	conv.	3 × 3	2 × 2	1 × 1	256
5-12	conv.	3 × 3	1 × 1	1 × 1	256
13	deconv.	4 × 4	$\frac{1}{2} \times \frac{1}{2}$	1 × 1	128
14	conv.	3 × 3	1 × 1	1 × 1	128
15	conv.	4 × 4	1 × 1	1 × 1	64
16	conv.	3 × 3	1 × 1	1 × 1	32
17	output	3 × 3	1 × 1	1 × 1	3

2.2. Inpainting network

The inpainting network is based on the Context Encoder network [6]. The completed edge map, along with the masked input, is first mapped to hidden representations through the encoder. The decoder takes the latent feature representation to predict the missing region.

The inpainting network is trained by regressing to the ground truth content of the missing region. Reconstruction loss and adversarial loss are used together to handle the similarity in the overall structure and naturalness of the generated missing region. Rather than taking GAN loss [20], we use the Wasserstein GAN (WGAN) loss [21] due to its ease of training and good results. For each pair of image (x, y), x is the missing region and y is the combination of corresponding masked region and the completed edge map. The inpainting network F produces a predicted missing region $F(y)$, and D is the adversarial discriminative model.

The objective for discriminator is the likelihood whether the input is real or predicted one:

$$\mathcal{L}_a = \max_{D \in L} \mathbb{E}_{x \in \mathcal{X}}[D(x) \text{ - } D(F(y))] \quad (1)$$

where L is the set of 1-Lipschitz functions. $\mathcal{X}$ represents the distributions of real data x.

3. EXPERIMENTAL RESULTS

We now evaluate the proposed E-CE model for image inpainting. The datasets and experimental settings are firstly introduced, followed by visualized results.

3.1. Datasets and experimental settings

Datasets. We experiment with images from two datasets: 137K Amazon Handbag images with 138,767 handbag images [15] and 100K-ImageNet [16] with 1,260,000 images from 1000 classes. To save computational time in this work, we use 100,000 images chosen at random in [6]. In the former dataset, the environment is simple and clear. We use this dataset to verify whether edges can formalize the shape of the completed objects. The latter dataset contains complex environment, which is used to test the overall performance of the proposed method. Only images from the datasets without any of the accompany labels are used. All images are cropped and resized to 128×128 with 64×64 region in the center masked out. 300 images out of each dataset are removed for testing while left are used for training.

Baselines. We compare our method with both local information based approaches and learning based approaches. The selected approaches in the former class are total variation (TV) approach [3] and Exemplar-Based approach (EB) [22]. The first learning based baseline is CE [6]. Due to the reason that the discriminators in our method are trained with WGAN loss, we also compare our method with the WGAN loss based CE implemented (CE-W).

Implementation details. We implement this network using pytorch [23] toolbox. We used batch size 64 in all experiments. The training of E-CE is separated into two stages. In the first stage, we train the edge completion network independently. The network is initialized at random, and trained end-to-end with reconstruction loss (L2) and adversarial loss. To deal with the edges on the boundaries of the mask, we copy the context (by 4px) outside the hole to the inner boundary and the filling position are masked out again After edge extraction. In the second stage, we train the modified CE part for image inpainting combine the masked input and the edge map. The training of edge completion network and inpainting network follow the training procedure proposed in WGAN [21] and apply the RMSProp solver [24]. We keep the dimension of the latent vector to 4000, the same as used in the original CE. The weights for reconstruction loss and adversarial loss are set to 0.99 and 0.01 respectively.

3.2. Results

The visualized results are shown in Fig. 3 and Fig. 4 for both datasets. In general, classical methods (TV, EB) can only produce blurred or cracked results in the tasks of completing large missing region. Results from context based approaches are more semantically correct than classical methods. Our proposed method generates better results than CE and CE-W in the sense of precise object shapes and structures.

In the case of clear environment as in Handbag dataset, our edge completion network is able to draw the boundary of the object (Fig. 3 (d)). We can also notice that the final completed images (Fig. 3. (i)) from our method can fit well with the edge maps, verified that the edge maps have strong effects on the texture prediction.

When processing ImageNet dataset with complex environment, our proposed method can successfully avoid texture penetration of neighboring object (e.g. row 1-3 in Fig.4). We attribute it to the edge map, which recovered the boundary between objects in the edge completion step. We also present a negative sample of ImageNet in the fifth row of Fig. 4. Little fine structures in the face area are generated in the edge completion step, leading to a weird dog face after inpainting. This is due to lack of context information inferred from its known region, which can also be seen in the results from CE and CE-W.

Table 2 reports the quantitative results of completed region on 100K-ImageNet dataset, which reveals similar trend with the qualitative results. The PSNR value of our methods is about 0.39 dB higher than CE-W.

Table 2. Numerical comparison on 100K-ImageNet dataset.

Method	Mean L1 loss	Mean L2 loss	PSNR	SSIM
CE	26.41%	14.91%	15.13 dB	0.495
CE-W	24.19%	13.16%	15.68 dB	0.515
Our method	**22.50%**	**12.19%**	**16.07** dB	**0.549**

4. CONCLUSION

In this paper, we proposed an edge-aware image inpainting method to handle deformed shapes in the previous learning based inpainting approaches. Moreover, according to the characteristics of binary edges, we developed an edge completion network. Compared to CE, our method can obtain images with better structures and correctly located textures. Experimental results demonstrated its superior performance on challenging image inpainting examples. In the future work, we will further look into the problems of how to generalize the edge-aware context encoder to applications lying in the knowledge and object domain, such as knowledge-based coding [25] and person re-identification [26].

Acknowledgements. This work was supported by National Natural Science Foundation of China (U1736206, 61231015, 61502348, 61671336, 61671332), National High Technology Research and Development Program of China (2015AA016306), EU FP7 QUICK project under Grant Agreement (PIRSES-GA-2013-612652), Hubei Province Technological Innovation Major Project （2017AAA123）.

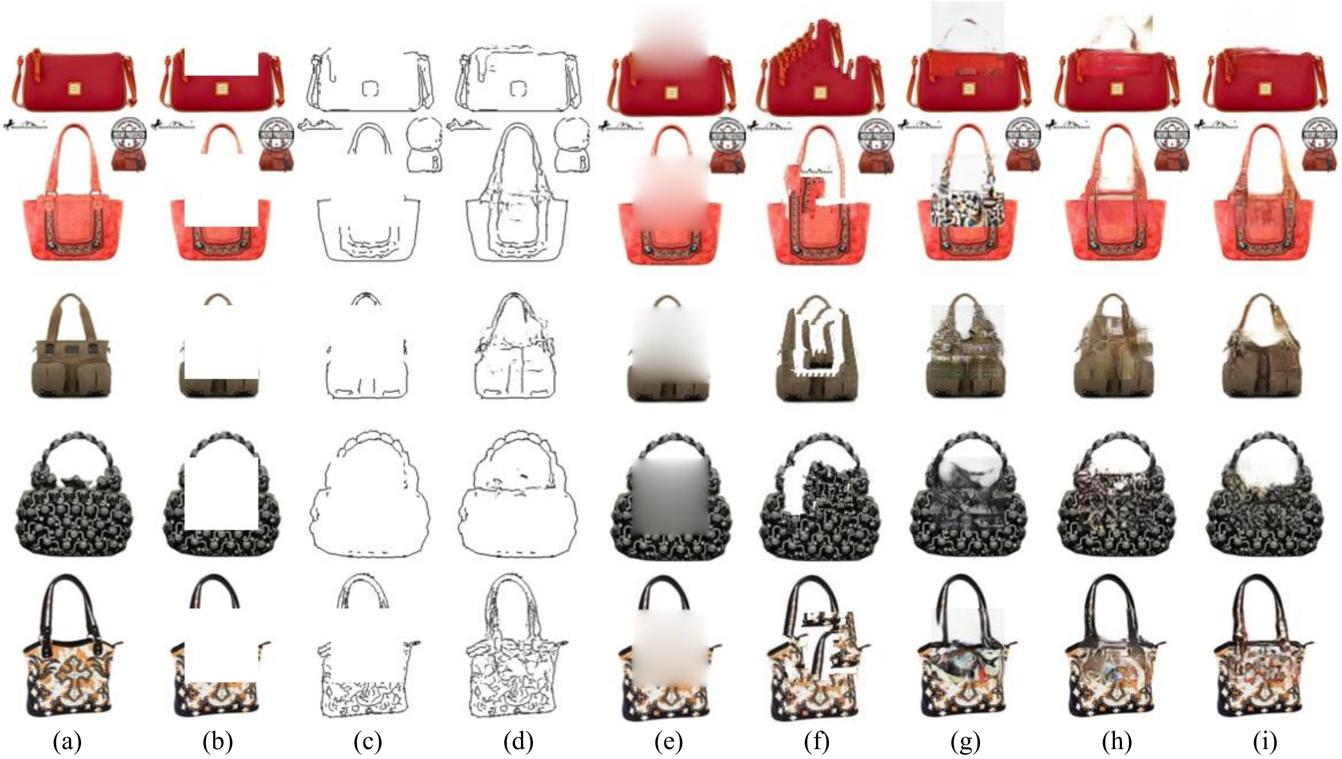

Fig. 3. Inpainting results for Handbag dataset. (a) original image; (b) masked image; (c) extracted edges; (d) completed edge map; (e) TV; (f) EB; (g) CE; (h) CE-W; (i) Our method.

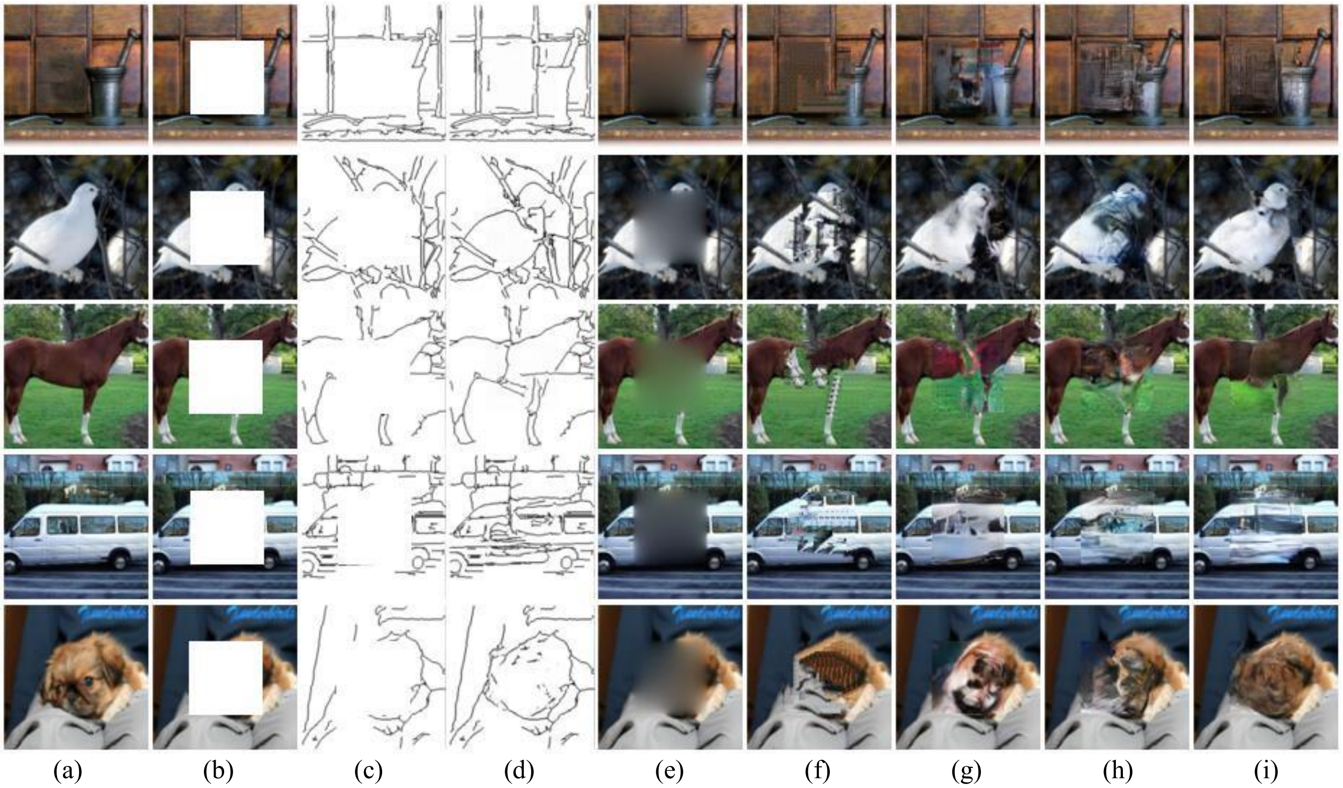

Fig. 4. Inpainting results for 100K-ImageNet dataset. (a) original image; (b) masked image; (c) extracted edges; (d) completed edge map; (e) TV; (f) EB; (g) CE; (h) CE-W; (i) Our method.

5. REFERENCES

[1] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, “Image inpainting,” In *Proc. ACM Conf. Comp. Graphics (SIGGRAPH)*, pp: 417–424, 2000.

[2] J. Weickert, “Theoretical foundations of anisotropic diffusion in image processing,” *Comput. Suppl.*, vol. 11, pp: 221–236, 1996.

[3] L. Rudin, S. Osher, and E. Fatemi, “Nonlinear total variation based noise removal algorithms,” *Phys. D*, vol. 60, pp: 259–268, 1992.

[4] L. Wei and M. Levoy, “Fast texture synthesis using tree-structured vector quantization,” In *Proc. ACM Conf. Comp. Graphics (SIGGRAPH)*, pp: 479–488, 2000.

[5] Z. Xu and J. Sun, “Image inpainting by patch propagation using patch sparsity,” In *IEEE Transactions on Image Processing*, vol. 19, no. 5, pp: 1153–1165, 2010.

[6] D. Pathak, P. Krähenbühl, J. Donahue, T. Darrell and A. A. Efros, "Context Encoders: Feature Learning by Inpainting," In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp: 2536-2544, 2016.

[7] Y. Li, S. Liu, J. Yang, and M.-H. Yang, “Generative Face Completion,” In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* pp: 3911-3919, 2017.

[8] S. Iizuka, E. Simo-Serra, and H. Ishikawa, “Globally and locally consistent image completion,” In *ACM Transactions on Graphics*, vol. 36, no. 4, pp: 1-14, 2017.

[9] A. Dosovitskiy, T. Brox, “Generating images with perceptual similarity metrics based on deep networks,” In *Advances in Neural Information Processing Systems (NIPS)*, pp: 1-9, 2016.

[10] A. B. L. Larsen, S. K. Sønderby, H. Larochelle, O. Winther, “Autoencoding beyond pixels using a learned similarity metric,” In *Proceedings of the 33rd International Conference on International Conference on Machine Learning (ICML)*, pp: 1558-1566, 2016.

[11] Shu Zhang, Ran He, Tieniu Tan, “DeMeshNet: Blind Face Inpainting for Deep MeshFace Verification,” *arXiv preprint arXiv:1611.05271*, 2016.

[12] C. Yang, X. Lux, Z. Liny, E. Shechtmanz, O. Wang, and H. Lik, “High-Resolution Image Inpainting using Multi-Scale Neural Patch Synthesis,” In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp: 6721-6729, 2017.

[13] H. Wu, S. Zheng, J. Zhang, K. Huang, “GP-GAN: Towards Realistic High-Resolution Image Blending”, *arXiv preprint arXiv:1703.07195*, 2017.

[14] R. Gao, K. Grauman, “On-Demand Learning for Deep Image Restoration,” In *IEEE International Conference on Computer Vision (ICCV)*, pp: 1095-1104, 2017.

[15] J.-Y. Zhu, P. Krähenbühl, E. Shechtman, and A. A. Efros, “Generative visual manipulation on the natural image manifold,” In *European Conference on Computer Vision (ECCV)*, pp: 597-613, 2016.

[16] 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,” In *International Journal of Computer Vision*, vol. 115, no. 3, pp: 211-252, 2014.

[17] J. Canny, “A Computational Approach to Edge Detection,” In *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp: 679–698, 1986.

[18] S. Xie and Z. Tu. “Holistically-nested edge detection,” In *IEEE International Conference on Computer Vision (ICCV)*, pp: 1395-1403, 2016.

[19] J. Long, E. Shelhamer, T. Darrell. “Fully convolutional networks for semantic segmentation,” In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp: 3431-3440, 2015.

[20] A. Radford, L. Metz, And S. Chintala, “Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks,” In *International Conference On Learning Representations (ICLR)*, 2016.

[21] M. Arjovsky, S. Chintala, L. Bottou, “Wasserstein Gan,” *arxiv preprint arxiv:1701.07875*, 2017.

[22] A. Criminisi, P. Perez And K. Toyama, “Region Filling And Object Removal By Exemplar-Based Image Inpainting,” In *IEEE Transactions on Image Processing*, Vol. 13, No. 9, pp. 1200-1212, 2004.

[23] A. Paszke, S. Chintala, R. Collobert, K. Kavukcuoglu, C. Farabet, S. Bengio, I. Melvin, J. Weston, and J. Mariethoz, “PyTorch: Tensors and Dynamic Neural Networks in Python with Strong GPU Acceleration,” May 2017. http://pytorch.org

[24] Tieleman, T., Hinton, G.: Lecture 6.5rmsprop: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural Networks for Machine Learning (2012).

[25] J. Xiao, R. Hu, L. Liao, Y. Chen, Z. Wang and Z. Xiong, “Knowledge-based Coding of Objects for Multi-source Surveillance Video Data,” In *IEEE Transactions on Multimedia*, vol. 18, no. 9, pp: 1691-1706, 2016.

[26] Z. Wang, R. Hu, Y. Yu, J. Jiang, C. Liang, J. Wang, “Scale-Adaptive Low-Resolution Person Re-Identification via Learning a Discriminating Surface,” In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI)*, pp: 2669-2675, 2016.