Deep-learning-based image inpainting algorithms have shown great performance via powerful learned priors from numerous external natural images. However, they show unpleasant results for test images whose distributions are far from those of the training images because their models are biased toward the training images. In this paper, we propose a simple image inpainting algorithm with test-time adaptation named AdaFill. Given a single out-of-distributed test image, our goal is to complete hole region more naturally than the pre-trained inpainting models. To achieve this goal, we treat the remaining valid regions of the test image as another training cue because natural images have strong internal similarities. From this test-time adaptation, our network can exploit externally learned image priors from the pre-trained features as well as the internal priors of the test image explicitly. The experimental results show that AdaFill outperforms other models on various out-of-distribution test images. Furthermore, the model named ZeroFill, which is not pre-trained also outperforms the pre-trained models sometimes.

Index Terms— Image Inpainting, Internal Learning, Test-time Adaptation

2. RELATED WORKS

Natural images have a high unique internal similarity, where similar structures or textures across various scales appear re-
Fig. 2: Left: Overall flow. In the training phase, we begin from the pre-trained inpainting network $G$ (or random initialization for ZeroFill). Next, for the test-time adaptation, we degrade a test image $x_d$ with random child masks $M_{c,i}=0,1,2,...$ and put them into the network. The output of the network has to be same with the test image $x_d$ at the valid regions. After test-time training, the test image $x_d$ is passed along with its parent mask $M_p$ to get the final inpainted image. Right: Structure of the inpainting network $G$.

Currently within an image. Several studies have previously verified that such internal similarity can be utilized for the single image super-resolution task [6, 7, 8]. They show that the internal statistical prior from a single image is powerful and often better than the generalized statistics from the large-scale training.

Our work is closely related to the ZSSR [6] that perform image super-resolution from a single image via internal learning. They artificially generate training samples from a low-resolution test image using re-downsampling. Compared with the ZSSR that exploits whole degraded images, our method utilizes valid regions as strong training cues. Similarly, DIP [5] propose a method that implicitly learns the prior of a single image. They show that this internal prior can be used to recover images with various types of degradations including image inpainting. However, such implicit internal prior has difficulty in recovering extreme degradations such as large holes or holes with extreme non-local patterns. In contrast, our method explicitly learns the internal prior using artificial training samples as well as external prior using large-scale pre-training.

3. METHOD

Our overall framework and network structure are described in Fig. 2. In the training phase, we begin from the pre-trained inpainting network $G$ and we fine-tune on a single degraded image $x_d$. For ZeroFill, we skip the pre-training process. We assume that the image $x_d$ is distorted by a distortion function $d(\cdot)$ with a parent mask $M_p$ from the clean image $x$ like $x_d = d(x)$. The parent mask $M_p$ represents the invalid pixels of $x_d$ with value 1 and valid pixels of $x_d$ with value 0. Therefore, we can represent the distorted image as $x_d = x \odot (1 - M_p) + M_p$, where $\odot$ represents element-wise multiplication. To enable our network to learn the internal similarity and exploit it for inpainting, we define a similar distortion function $d'(\cdot)$ with the child mask $M_c$. As illustrated in Fig. 2 we degrade the given image using the child mask $d'(x_d) = x_d \odot (1 - M_c) + M_c$. The child masks are randomly generated during training. We denote this double-distorted image as $x_{dd'}$. In the case where the shape of the parent mask $M_p$ is different from the irregular mask or box mask, we use the parent mask as a child mask with random rotation and scaling at a certain rate.

Our goal is to make the network learn the mapping function from the double-distorted image $x_{dd'}$ to the given single distorted image $x_d$. 

$$\bar{x} = G([x_{dd'}, M_c])$$

where $\bar{x}$ is a preliminary prediction, $G$ is our inpainting network, and $[\cdot , \cdot]$ is concatenation operation. The predicted image $\bar{x}$ has to be same with the given image $x_d$ for the valid pixels. Since we do not know the ground truth of the invalid pixels in $x_d$, we degrade $\bar{x}$ with the same distortion function $\bar{x}_d = d(\bar{x})$. After degradation, we use following $L1$ loss to train the inpainting network $G$.

$$L_{L1} = |\bar{x}_d - x_d|_1$$

From this training step, the network $G$ can learn the restoration patterns in the degraded image using the valid regions while ignoring the parent distortions.
Table 1: Quantitative comparison with GatedConv [1], EdgeConnect [3], and DIP [5]

| Model         | Dataset   | PSNR  | SSIM  | LPIPS | PSNR  | SSIM  | LPIPS | PSNR  | SSIM  | LPIPS |
|---------------|-----------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
|               | T91       | 27.15 | 0.889 | 0.0755| 27.85 | 0.901 | 0.0692| 25.46 | 0.851 | 0.1047|
|               | Urban100  | 23.14 | 0.854 | 0.0722| 24.06 | 0.866 | 0.0721| 22.23 | 0.805 | 0.1082|
|               | Google map| 24.65 | 0.846 | 0.0939| 26.25 | 0.848 | 0.0888| 24.47 | 0.836 | 0.1164|
|               | Facade    | 26.25 | 0.900 | 0.0570| 26.02 | 0.886 | 0.0688| 25.79 | 0.890 | 0.0691|
|               | BCCD      | 34.25 | 0.956 | 0.0595| 34.43 | 0.954 | 0.0662| 30.47 | 0.948 | 0.0890|
|               | KLH       | 33.25 | 0.823 | 0.1162| 20.91 | 0.791 | 0.3419| 30.33 | 0.751 | 0.1579|
|               | Document  | 19.67 | 0.910 | 0.0762| 18.84 | 0.876 | 0.1316| 17.49 | 0.865 | 0.1122|

4. EXPERIMENTS

Settings. We pre-train the network using Places365 [18] dataset for 1 epochs with the settings in [3]. For the test-time adaptation, we use the following hyper-parameters: batch size 8, learning rate 0.0001, Adam [19] optimizer with $\beta_1 = 0.5, \beta_2 = 0.9$, and 1,000 training iterations. For ZeroFill, we use 5,000 iterations without pre-training. We evaluate our model with LPIPS [16], SSIM, and PSNR. We compare our model with other models that learn only the explicit prior only (pre-trained models, GatedConv [1] and EdgeConnect [3]), and only the internal prior only (DIP [5]).

Dataset. We use various datasets for evaluating our model: T91 [9], Urban100 [10], Google Map [11], Facade [12], BCCD [13], KLH [14], BSD200 [20], and Document [15]. These are out-of-distributed from the Places365 [18] dataset whose distribution is focused on various places images. In contrast, these images are small objects, natural scenes, artificial structures, medical images, satellite images, or text images. We subsample and pre-process each dataset and use two type of holes: box mask and irregular mask. For detailed description, please refer to the supplementary materials.

4.1. Experimental Results

Quantitative results are described in Table 1. From these results, our model outperforms DIP for all datasets and metrics. In almost cases, our model is superior to the pre-trained models even though our model is a one-stage network. It reveals that exploiting the internal statistics of a test image is critical to image inpainting. If the dataset has strong internal similarities, such as Urban100, Google Map, and Facade, our model consistently performs better. In addition, if the distribution of the dataset is far from that of the training dataset, such as KLH and Document, pre-trained models cannot recover well.

Our qualitative results are compared in Fig. 4. As mentioned above, similar results are observed. In the case of large internal similarity within an image, our model perfectly recover the hole regions, while other models show severe artifacts.

4.2. Internal Similarity

From the Fig. 3 result show that the higher internal similarity score, the better restoration performance is achieved from our method. Our method perfectly recovers the hole regions when the internal similarity is high (the first column of Fig. 3). To get the internal similarity, we first use pre-trained VGG19 [21] and extract features from relu 5-1 layer. Next, we calculate the pixel-wise similarities using the cosine similarity to get the similarity map of size $HW \times HW$, where $H$ and $W$ are the height and width of the feature map, respectively. Finally, we average the similarity map to get the final internal similarity score.

4.3. Ablation Study

We conducted ablation studies to find the optimal structure for the test-time adaptation with a single image. These results
Fig. 4: Qualitative comparison results with pre-trained model of GatedConv [1], EdgeConnect [3] and DIP [5]. We can confirm that pre-trained models show color artifacts and lower ability in capturing internal similarity in a single image.

Table 2: Ablation study. PT: pre-training, TTA: test-time adaptation, One St: one-stage network, BN: batch normalization, NN: nearest-neighbor upsampling with convolution.

| Model  | PT | TTA | One St | BN, NN | PSNR | SSIM | LPIPS |
|--------|----|-----|--------|--------|------|------|-------|
| EC     | √  | √   | √      |       | 25.63| 0.847| 0.1283|
| EC-TTA | √  | √   | √      |       | 28.52| **0.884** | 0.1095|
| AdaFill| √  | √   | √      |       | 28.57| 0.883| 0.0882|
| ZeroFill| √ | √   | √      |       | 27.47| 0.878| 0.1108|

are compared in Table 2. For the ablation experiments, we use the first 10 images from each dataset in Table 1. We modified two things from the EdgeConnect [3] baseline structure. The first one is using only the second stage of the EdgeConnect to reduce the number of parameters. The second one is replacing instance normalization [22] + transposed convolution with batch normalization [23] + nearest-neighbor upsampling with convolution. These modifications increase the perceptual restoration quality, and reduce the color and annoying artifacts a lot. The results also show that our non-pre-trained model, ZeroFill even show a slightly better performance than the pre-trained model.

5. CONCLUSION

We propose a simple test-time adaptation scheme called AdaFill for image inpainting and ZeroFill as an unsupervised version. The results show that the previous pre-trained models cannot generalize well on the out-of-distributed images. In contrast, our methods can overcome this domain gap and fully exploit the internal similarity of test images. As a future works, to reduce the test time, exploiting meta-learning [24] can be adapted for practical usage.

Acknowledgement. This research was supported by R&D program for Advanced Integrated-intelligence for Identification (AIID) through the National Research Foundation of KOREA(NRF) funded by Ministry of Science and ICT (NRF-2018M3E3A1057289).
6. REFERENCES

[1] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang, “Free-form image inpainting with gated convolution,” in Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 4471–4480.

[2] Guilin Liu, Fitsum A Reda, Kevin J Shih, Ting-Chun Wang, Andrew Tao, and Bryan Catanzaro, “Image inpainting for irregular holes using partial convolutions,” in Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 85–100.

[3] Kamyar Nazeri, Eric Ng, Tony Joseph, Faisal Z Qureshi, and Mehran Ebrahimi, “Edgeconnect: Generative image inpainting with adversarial edge learning,” arXiv preprint arXiv:1901.00212, 2019.

[4] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang, “Generative image inpainting with contextual attention,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 5505–5514.

[5] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky, “Deep image prior,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 9446–9454.

[6] Assaf Shocher, Nadav Cohen, and Michal Irani, ““zero-shot” super-resolution using deep internal learning,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 3118–3126.

[7] Maria Zontak and Michal Irani, “Internal statistics of a single natural image,” in CVPR 2011. IEEE, 2011, pp. 977–984.

[8] Daniel Glasner, Shai Bagon, and Michal Irani, “Super-resolution from a single image,” in 2009 IEEE 12th international conference on computer vision. IEEE, 2009, pp. 349–356.

[9] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang, “Deep laplacian pyramid networks for fast and accurate super-resolution,” in IEEE Conference on Computer Vision and Pattern Recognition, 2017.

[10] J. Huang, A. Singh, and N. Ahuja, “Single image super-resolution from transformed self-exemplars,” in 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 5197–5206.

[11] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros, “Image-to-image translation with conditional adversarial networks,” CVPR, 2017.

[12] Radim Tyleček and Radim Šára, “Spatial pattern templates for recognition of objects with regular structure,” in Proc. GCPR, Saarbrucken, Germany, 2013.

[13] akshaylamba cosmoscad, “Blood cell count and detection dataset,”.

[14] “Automatic particle selection: results of a comparative study,” 2004, vol. 145, pp. 3 – 14, Automated Particle Selection for Cryo-Electron Microscopy.

[15] National Institute of Standards and Technology, “nist special database 2,” 2009.

[16] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang, “The unreasonable effectiveness of deep features as a perceptual metric,” in CVPR, 2018.

[17] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer, “Automatic differentiation in pytorch,” in NIPS-W, 2017.

[18] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba, “Places: A 10 million image database for scene recognition,” IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017.

[19] Diederik P Kingma and Jimmy Ba, “Adam: A method for stochastic optimization,” arXiv preprint arXiv:1412.6980, 2014.

[20] D. Martin, C. Fowlkes, D. Tal, and J. Malik, “A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics,” in Proc. 8th Int’l Conf. Computer Vision, July 2001, vol. 2, pp. 416–423.

[21] Karen Simonyan and Andrew Zisserman, “Very deep convolutional networks for large-scale image recognition,” arXiv preprint arXiv:1409.1556, 2014.

[22] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky, “Instance normalization: The missing ingredient for fast stylization,” arXiv preprint arXiv:1607.08022, 2016.

[23] Sergey Ioffe and Christian Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” arXiv preprint arXiv:1502.03167, 2015.

[24] Jae Woong Soh, Sunwoo Cho, and Nam Ik Cho, “Meta-transfer learning for zero-shot super-resolution,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 3516–3525.
7. SUPPLEMENTARY MATERIALS

7.1. Experimental Configuration

**Dataset:** For larger than $256 \times 256$ images, we resize images to 256 pixels for shorter axis and random crop to make final resolution of $256 \times 256$. We only use first 100 images for dataset which is set of more than 100 images. In case of Document [13] dataset, we just downsample by a factor of 4 and random crop to get the resolution of $256 \times 256$.

**Mask:** We use irregular mask [1] with rate of 10 to 30%. In case of random box mask, we use 5 to 15% rate. We average results from irregular and random box mask to get final result value.

7.2. Results and Comparisons

7.2.1. **BSD200 [20] Dataset**

![Fig. 5: Qualitative comparison results with BSD200 [20] dataset](image)
7.2.2. General100 Dataset

![Fig. 6: Qualitative comparison results with General100 dataset](image)

7.2.3. T91 Dataset

![Fig. 7: Qualitative comparison results with T91 dataset](image)
7.2.4. Urban100 [10] Dataset

Fig. 8: Qualitative comparison results with T91 [9] dataset

Fig. 9: Qualitative comparison results with Urban100 [10] dataset
### 7.2.5. Google Map Dataset

| GT | Masked | GC | EC | DIP | Ours |
|----|--------|----|----|-----|------|
| ![Image](image1.png) | ![Image](image2.png) | ![Image](image3.png) | ![Image](image4.png) | ![Image](image5.png) | ![Image](image6.png) |

**Fig. 10:** Qualitative comparison results with Urban100 dataset

| GT | Masked | GC | EC | DIP | Ours |
|----|--------|----|----|-----|------|
| ![Image](image1.png) | ![Image](image2.png) | ![Image](image3.png) | ![Image](image4.png) | ![Image](image5.png) | ![Image](image6.png) |

**Fig. 11:** Qualitative comparison results with Google Map dataset
7.2.6. Facade [12] Dataset

**Fig. 12:** Qualitative comparison results with Google Map [11] dataset

**Fig. 13:** Qualitative comparison results with Facade [12] dataset
7.2.7. BCCD [13] Dataset

**Fig. 14:** Qualitative comparison results with Facade [12] dataset

**Fig. 15:** Qualitative comparison results with BCCD [13] dataset
7.2.8. KLH [14] Dataset

Fig. 16: Qualitative comparison results with KLH [14] dataset

7.2.9. Document [15] Dataset

Fig. 17: Qualitative comparison results with Document [15] dataset
8. REFERENCES

[1] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang, “Free-form image inpainting with gated convolution,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 4471–4480.

[2] Guilin Liu, Fitsum A Reda, Kevin J Shih, Ting-Chun Wang, Andrew Tao, and Bryan Catanzaro, “Image inpainting for irregular holes using partial convolutions,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 85–100.

[3] Kamyar Nazeri, Eric Ng, Tony Joseph, Faisal Z Qureshi, and Mehran Ebrahimi, “Edgeconnect: Generative image inpainting with adversarial edge learning,” *arXiv preprint arXiv:1901.00212*, 2019.

[4] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang, “Generative image inpainting with contextual attention,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 5505–5514.

[5] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky, “Deep image prior,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 9446–9454.

[6] Assaf Shocher, Nadav Cohen, and Michal Irani, “‘zero-shot’ super-resolution using deep internal learning,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 3118–3126.

[7] Maria Zontak and Michal Irani, “Internal statistics of a single natural image,” in *CVPR 2011*, IEEE, 2011, pp. 977–984.

[8] Daniel Glasner, Shai Bagon, and Michal Irani, “Super-resolution from a single image,” in *2009 IEEE 12th international conference on computer vision*. IEEE, 2009, pp. 349–356.

[9] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsian Yang, “Deep laplacian pyramid networks for fast and accurate super-resolution,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[10] J. Huang, A. Singh, and N. Ahuja, “Single image super-resolution from transformed self-exemplars,” in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 5197–5206.

[11] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros, “Image-to-image translation with conditional adversarial networks,” *CVPR*, 2017.

[12] Radim Tyleček and Radim Šára, “Spatial pattern templates for recognition of objects with regular structure,” in *Proc. GCPR*, Saarbrucken, Germany, 2013.

[13] akshaylamba cosmicad, “Blood cell count and detection dataset,” .

[14] “Automatic particle selection: results of a comparative study,” 2004, vol. 145, pp. 3 – 14, Automated Particle Selection for Cryo-Electron Microscopy.

[15] National Institute of Standards and Technology, “nist special database 2,” 2009.

[16] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang, “The unreasonable effectiveness of deep features as a perceptual metric,” in *CVPR*, 2018.

[17] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer, “Automatic differentiation in pytorch,” in *NIPS-W*, 2017.

[18] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba, “Places: A 10 million image database for scene recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[19] Diederik P Kingma and Jimmy Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.

[20] D. Martin, C. Fowlkes, D. Tal, and J. Malik, “A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics,” in *Proc. 8th Int’l Conf. Computer Vision*, July 2001, vol. 2, pp. 416–423.
[21] Karen Simonyan and Andrew Zisserman, “Very deep convolutional networks for large-scale image recognition,” arXiv preprint arXiv:1409.1556, 2014.

[22] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky, “Instance normalization: The missing ingredient for fast stylization,” arXiv preprint arXiv:1607.08022, 2016.

[23] Sergey Ioffe and Christian Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” arXiv preprint arXiv:1502.03167, 2015.

[24] Jae Woong Soh, Sunwoo Cho, and Nam Ik Cho, “Meta-transfer learning for zero-shot super-resolution,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 3516–3525.