Photo-Realistic and Robust Inpainting of Faces Using Refinement GANs

Dejan Malesevic, Christoph Mayer, Shuhang Gu, and Radu Timofte

1 Introduction

Image inpainting tackles the problem of filling missing or replacing damaged parts of an image. For a large extent of the missing region reconstructing the original content is almost impossible. Therefore, the goal is to inpaint new parts that are visually appealing, such that they fit the surrounding content and that the whole image looks realistic.

Image inpainting is an important problem in digital image processing [10]. It allows to restore images that are damaged by scratches, holes or small artifacts. Furthermore, it enables to erase overlaid text or logos from the image. Similarly, image inpainting allows filling gaps caused by object removal [5] during image editing. Other possible applications are censoring [11, 12] and decensoring. Such as restoring blurred faces or license numbers or decensoring images by inpainting fake but visually appealing faces or license numbers instead of blurring them. Furthermore, inpainting can replace image parts that were lost during data transmission. Hence, typical patterns to be restored are: many fine grained structures (lost during transmission or overlaid text) and small to large polygons covering large parts of the image (removing objects, logos or decensoring and censoring images).

Image inpainting is an ill-posed problem, such that no unique solution exists. Therefore, it requires priors that depend on a consensus assumption that surrounding regions share the same statistic of properties than the gap. Enforcing these priors during restoration aims for inpainting regions that are conclusive and visually appealing.

D. Malesevic · C. Mayer (✉) · S. Gu · R. Timofte
Computer Vision Laboratory, ETH Zürich, Switzerland
e-mail: madejan@student.ethz.ch; chmayer@vision.ee.ethz.ch; shuhang.gu@vision.ee.ethz.ch; radu.timofte@vision.ee.ethz.ch

S. Escalera et al. (eds.), *Inpainting and Denoising Challenges*,
The Springer Series on Challenges in Machine Learning,
https://doi.org/10.1007/978-3-030-25614-2_11

Filling small artifacts such as holes, scratches or overlaid text is less challenging because the extent of the gap is small and already simple properties lead to sufficient information to plausibly fill the holes. Typical priors base upon smoothness assumptions using the total variation in spatial domain or sparsity in curvelet or wavelet domain [3, 4, 18, 21]. Patch Match (PM) [2] searches for patches with similar properties in the image, that enable inpainting the missing parts. Dictionary learning, first trains statistically representative patterns on a dataset containing similar images, such that the clean pixels and the learned elements allow reconstructing the image [17, 24]. Although dictionary learning based methods lead to higher quality restorations than using sparsity or smoothness, they fail in filling missing parts with a significantly larger extent. Instead of filling small holes, such large gaps lead to more difficult tasks of semantic inpainting. The goal is predicting high quality content of a large region based on the surrounding pixels and the global scene that may be far from the lost original content but looks plausible and appealing.

Most methods that effectively tackle such tasks require large problem specific datasets, use adversarial training and generative models [6, 13, 19, 23], such as Variational Auto Encoderss (VAEs) [15] or Generative Adversarial Networkss (GANs) [7]. Such generative models approximate the data distribution and allow to produce synthetic images at a high visual quality. Thus, the key idea is using such models to generate images that achieve a consensus between the clean part of the damaged and the synthetic image and inpaint the missing regions. Pathak et al. [19] use a VAE for image inpainting. The encoder maps a damaged image to the latent space and the decoder generates a synthetic image from the latent representation. During training a consensus loss ensures that the clean parts of the damaged and synthetic images agree and an adversarial loss ensures that the restored image have a high quality. This method respects the hole patterns only during training and not at inference. Hence, the visual quality of the reconstructions is less convincing.

Conversely, Yeh et al. use a GAN for image inpainting. Their method requires solving a minimization problem to find the corresponding latent representation instead of predicting it with an encoder. An image specific loss based on the clean pixels in the vicinity of holes and a dataset specific perceptual loss, lead to high quality image inpainting. After inpainting, Poisson blending flawlessly fuses the inpainted and clean pieces to the final image.

1.1 Contributions

In this paper we propose an new method for image inpainting using a refiner based on a GAN. Figure 1 shows a sample image where simple image inpainting produces unaesthetic and unrealistic results, whereas our refinement improves the restoration significantly. The novelty and the advantage of the refiner is its global consistency loss that requires a global consensus between all clean pixels and the corresponding synthetic parts. The refiner uses an already plausible inpainting from a former stage and therefore converges faster and ensures that the restored image and the synthetic

Fig. 1 Sample image that demonstrates the deficits of semantic image inpainting that our proposed refinement method can correct

image are closer. Closer images are beneficial as the perceptual loss assesses the quality of restored images via the synthetic image. Therefore, a small perceptual loss for a synthetic image that is largely different than the restored image has a limited expressiveness for the image quality. As our refinement method is similar to semantic image inpainting developed by Yeh et al. [23], we will briefly introduce the corresponding components and highlight its shortcomings and the differences to our refinement approach. To illustrate that our method overcomes these deficits, we use the CelebA [16] face dataset containing more than 200k face images. Humans are most sensitive even for small artifact in faces. Thus, we use face images to evaluate our method because restoring missing parts of a face, is among the most challenging problems in image inpainting.

Contributions

- Disentangling the different components of a state-of-the-art image inpainting method, showing its deficits and how to overcome them.
- Developing of a refinement GAN for improving image inpainting using a global consensus loss.
- Proposing and implementing of a pipeline that benefits from our refinement GAN.
- Training and evaluating the image inpainting pipeline on face images using four classical metrics in image inpainting for various different distortion patterns.

1.2 Semantic Image Inpainting

Let $\mathcal{P} = \mathcal{M} \cup \mathcal{M}^c$, where $\mathcal{P}$ is the set that contains all the pixel indices of an image. An index of the set $\mathcal{M}$ corresponds to a clean pixel. The set $\mathcal{M}^c$ contains the indices of damaged or missing pixels and is therefore the complement of the set $\mathcal{M}$. Thus, we refer to the set, that contains all clean pixels, as $\mathcal{I}_{\mathcal{M}} = \{\mathcal{I}_i\}_{i \in \mathcal{M}}$. See Fig. 2 for an illustration of the introduced nomenclature using an example image.

In other words, image inpainting tries to predict $\mathcal{I}_{\mathcal{M}^c}$ using the damaged image $\mathcal{I}$ and the mask $\mathcal{M}$. Due to the ill-posedness of image inpainting, recovering the exact solution is usually impossible therefore, semantic image inpainting uses a GAN to

Fig. 2 Illustration of the introduced sets and which pixels they contain when applied to a sample image

hallucinate a plausible prediction of $\mathcal{I}_{\mathcal{M}^c}$. We train the GAN using the classical GAN objective, see Eq. (1), where G is the generator and D the discriminator that returns a probability estimate how likely the seen sample is real and the origin of the image lies in the true data distribution.

$$\min_G \max_D \; \mathbb{E}_{x \sim p_{\text{data}}}\Big[\log D(x)\Big] + \mathbb{E}_{z \sim p_z}\Big[\log\Big(1 - D\big(G(z)\big)\Big)\Big], \tag{1}$$

The generator of a GAN generates random synthetic samples of the data distribution for a sample, drawn from a fixed prior distribution. The goal of inpainting is using the generator to produce an image that agrees on the clean pixels and produces plausible content for the missing region. Instead of exhaustively drawing samples from the generator until one matches the requirements. Yeh et al. [23] proposed to define a loss, that measures the quality of the inpainting, and to minimize this loss using gradient decent and the chain rule to calculate the gradients. The optimization problem reads as follows

$$\begin{aligned} &\text{minimize} \quad \mathcal{L}_{\text{inpainting}}\,(x \mid y, \mathcal{S}) \\ &\text{subject to} \quad x = G(z), \end{aligned} \tag{2}$$

where x and y are the generated and the damaged image, $\mathcal{S}$ is a subset of the mask $\mathcal{M}$ that we will define shortly (see Fig. 2 for an example), $G(\cdot)$ is the generator and z the latent variable. Yeh et al. [23] proposed a loss function that consists of a content loss and a perceptual loss, see Eq. (3).

$$\mathcal{L}_{\text{inpainting}}(x|y, \mathcal{S}) = \mathcal{L}_{\text{content}}(x|y, \mathcal{S}) + \lambda\, \mathcal{L}_{\text{perception}}(x) \tag{3}$$

The content loss tries to achieve an agreement of clean pixels between the generated and the damaged image. However, Yeh et al. [23] proposed to consider only pixels that are in the vicinity of holes, instead of respecting all unimpaired pixels. The heuristic is, that pixels next to a hole contain far more information used to fill holes than pixels far away. Furthermore, their proposed loss contains a scalar that assigns a clean pixel a weight depending on the number of holes in local neighborhood, based on the same heuristic.

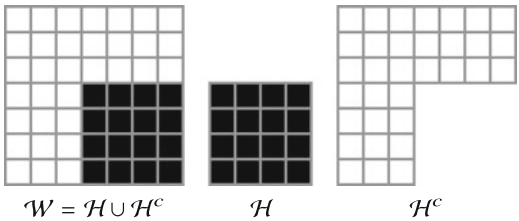

Fig. 3 Illustration of the sliding window set $\mathcal{W}$ and the two sets containing all indices corresponding to clean pixels ($\mathcal{M}$) or holes ($\mathcal{M}^c$). Black corresponds to holes and white to clean pixels

$$\mathcal{L}_{\text{content}}(x|y, \mathcal{S}) = \sum_{i \in \mathcal{S}} w_i \cdot |x_i - y_i|, \text{ where } w_i = \frac{|\mathcal{H}_i|}{|\mathcal{H}_i^c|} \tag{4}$$

Equation (4) shows the detailed content loss, where x and y are the generated and the damaged image, z is the latent variable, $\mathcal{S}$ is the set containing the selected pixels and w is a weight.

The weight corresponds to the ratio of holes $|\mathcal{H}_i|$ and clean pixels $|\mathcal{H}_i^c|$ in a centered $n \times n$ window $\mathcal{W}_i$ around the corresponding pixel i, see Fig. 3 for an illustration of the different sets. The expression $|\cdot|$ applied to a set denotes the cardinality, i.e. the number of element in the set. Note, that the weight for a clean pixel is zero, if all other pixels in the window are undamaged. Hence, the set $\mathcal{S}$ contains the indices of all pixels that contain at least one hole in their neighborhood. Thus, the set $\mathcal{S}$ used in $\mathcal{L}_{\text{content}}$ is a subset of the mask $\mathcal{M}$, such that $\mathcal{S} \subset \mathcal{M}$.

The idea of the perception loss is: favouring visually plausible synthetic images and penalizing unrealistic restorations. Fortunately, GANs consist beside the generator of a discriminator. The task of the discriminator is to identify whether an image is real or synthetic. In addition, the generator tries to fool the discriminator by producing images that the discriminator predicts as real although they are fake. Therefore, the perception loss uses the discriminator to assess whether the input image is real or fake. The loss is smaller the higher the probability of the discriminator that an images is real, see Eq. (5).

$$\mathcal{L}_{\text{perception}}(x) = \log\big(1 - D(x)\big) \tag{5}$$

Figure 4 shows a block diagram of the aforementioned semantic image inpainting method. The block requires the damaged input image y and the set $\mathcal{S}$ to select the pixel required to compute the content loss. Minimizing the content and the perception loss produces the generate image x_g.

1.3 Poisson Blending

Although the content loss favors a consensus on selected clean pixels in the vicinity of holes, there is no guarantee, that all pixels corresponding to the clean parts agree. Therefore, we replace the gap in the damaged image with the corresponding pixels from the synthetic image. To improve the quality of the fused image, Yeh et al. [23]

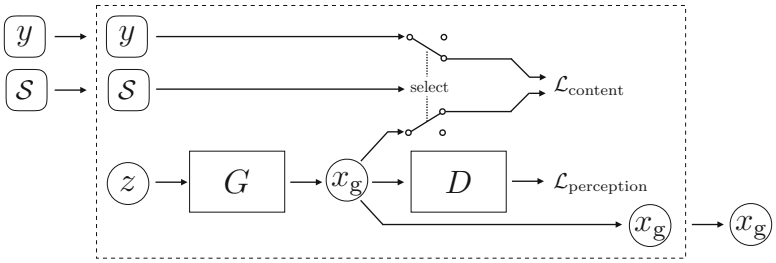

Fig. 4 Semantic image inpainting GAN. The method requires the damaged image y and the set S that selects the pixels used to compute the content loss. G and D are the generator and discriminator. We minimize the losses with respect to the latent variable z and the corresponding generated image x_g is the desired result

propose using Poisson blending to polish the reconstruction by smoothing the transition between clean and inpainted pixels, see Fig. 5 for examples. Poisson blending requires solving the optimization problem depicted in Eq. (6), where D is a linear discrete differentiation operator that approximates the gradient, $\mathcal{M}$ is the mask, x is the restored blended image, y is the damaged image and x_g is the generated image.

$$\begin{aligned} &\text{minimize} \quad \| \mathrm{D}\, x - \mathrm{D}\, x_g \|_2^2 \\ &\text{subject to } x_i = y_i, \quad \forall i \in \mathcal{M} \end{aligned} \tag{6}$$

Simply speaking, Poisson blending retrieves an image that achieves a consensus for pixels corresponding to the clean parts and the gradient between the synthetic and blended image should be as close as possible. Therefore, visible artifacts at the transition between the clean and inpainted region are penalized.

1.4 Shortcomings of Semantic Image Inpainting Without Refinement

The aforementioned semantic image inpainting has two different deficits. One is caused by the locally restricted content loss, that considers only the immediate vicinity of holes to condition the reconstruction and the other by the perception loss, that classifies the synthetic and not the restored image upon plausibility. This is especially harmful as the synthetic image seeks agreement with the damaged image only on a small region. Therefore, the synthetic image might look by itself realistic and might achieve a consensus on the selected region. However, transferring the inpainted region to the damaged image might lead to an implausible result. See Fig. 6, that shows eight different images with different artifacts. Mainly caused by

Fig. 5 Sample images highlighting that blending improves the image quality, by removing artifacts at the transition of inpainted and clean pixels

disagreeing face attributes between the synthetic and the damaged image. The first row shows an image of a girl but the inpainted region belongs to an adult and on the right, an image of a woman where inpainting adds a moustache. The other rows show that inpainting fails to restore symmetrical properties such as eye color, skin color, cheekbones, beard, mouth shape, eyebrows or eyeglasses. The problem is that the local extent of the content loss leads to limited information such as gender or other face attributes. Therefore, the generator might produce an image depicting a male instead of a female to fill the gap. Although the synthetic image might look plausible the fused may not.

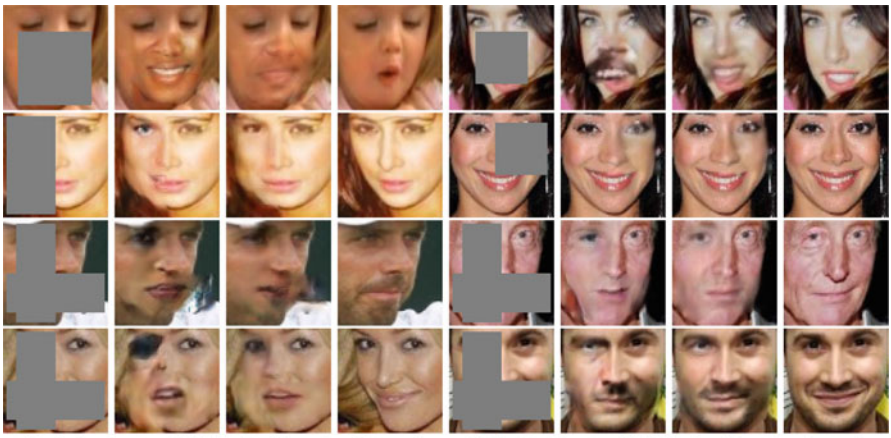

Fig. 6 Sample images where semantic image inpainting fails to produce visually appealing and plausible results. Conversely, our propose refinement method corrects many failures such as: removing moustache, enforcing similar eye and skin color, matching eye brows, cheekbones. In general our refinement leads to higher quality reconstructions that are more plausible and restores symmetric properties of a face

2 Improving Semantic Image Inpainting with Refinement GANs

As highlighted in Sect. 1.4, semantic image inpainting suffers from a too narrow field of view when conditioning the content loss and applying the perception loss on visually distinct images than the final reconstructions. Thus, we propose a refinement method, that takes all clean pixels into account and forces the damaged and the synthetic images to agree, such that the score of the perception loss of the synthetic image is more reliable.

Our refinement method operates similarly as semantic image inpainting. We solve an optimization problem by searching the values of the latent variable that corresponds to an image, minimizing our refinement objective. The objective consists of a perception loss and two content losses. The global consensus loss is minimal when the clean and the corresponding synthetic pixels agree. The inpainting loss favors similarly inpainted parts by the refiner as obtained from semantic inpainting with blending. We use the same perception loss as stated in Eq. (5). The two content losses read as follows, when using the previously introduced nomenclature,

$$\mathcal{L}_{\text{global-consensus}}(x|y, \mathcal{M}) = \sum_{i \in \mathcal{M}} |x_i - y_i| \tag{7}$$

$$\mathcal{L}_{\text{inpainting-consensus}}(x|y, \mathcal{M}^c) = \sum_{i \in \mathcal{M}^c} |x_i - y_i|, \tag{8}$$

where the only difference are the two distinct sets that contain the pixel positions. Equation (9) shows the full refinement loss with the two weighting parameters μ and λ.

$$\begin{aligned}\mathcal{L}_{\text{refinement}}(x|x_b, \mathcal{M}, \mathcal{M}^c) = &\mathcal{L}_{\text{global-consensus}}(x|x_b, \mathcal{M}) \\ &+ \mu\, \mathcal{L}_{\text{inpainting-consensus}}(x|x_b, \mathcal{M}^c) \\ &+ \lambda\, \mathcal{L}_{\text{perception}}(x)\end{aligned} \tag{9}$$

The parameter μ corresponds to the confidence or the plausibility of the inpainted region. Therefore, we are less confident for large gaps and use smaller values for μ the larger the damaged area. Note, that the proposed refinement loss considers all pixels of the inpainted and blended image instead of only a few clean pixels of the damaged image. Therefore, the loss is conditioned on the sets $\mathcal{M}$ and $\mathcal{M}^c$ and the inpainted and blended image x_b. λ regulates the additional cost added to the content loss for unrealistic reconstructions.

Figure 7 shows the block diagram of the proposed refinement GAN, where the selection process highlights that the refiner uses all pixels to compute the two content losses.

After refinement we propose again to use Poisson blending to smoothly fuse the inpainted region with the clean part. Figure 8 illustrates the full pipeline of the proposed semantic image inpainting with a refinement GAN.

The global awareness of clean and missing pixels of the refiner allows correcting errors of simple inpainting with blending. Figure 6 shows all kind of failures that the refiner eliminates. It restores symmetry such that eye color, skin color, eyebrows and cheekbones of the inpainted and clean parts agree. Furthermore, the refiner suppresses implausible face attributes such as mouth and nose of an adult for a child, moustache for a woman, female eye and lips for a man or inpainting eyeglasses for only one eye.

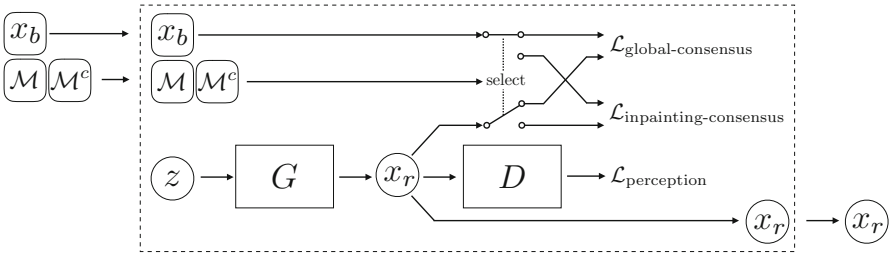

Fig. 7 Refinement GAN. The method requires an inpainted and blended image x_b and the set $\mathcal{M}$ and $\mathcal{M}^c$ to assign the pixels to the corresponding loss. G and D are the generator and discriminator. We minimize the losses with respect to the latent variable z and the corresponding refined image x_r is the desired result

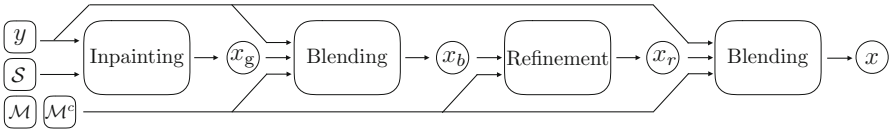

Fig. 8 Pipeline. The full pipeline consists of a cascade of an inpainting and refinement block. After each of this blocks we apply Poisson blending, to generate our final prediction of the restored image x from the damaged image y

3 Experiments and Results

3.1 Dataset

We use the CelebA dataset that contains more than 200k face images and use 2000 images each for validation and testing and use all other images for training. The images are aligned and cropped to $64 \times 64 \times 3$ pixels.

3.2 Implementation Details

We use the same GAN architecture and training policy as proposed by Yeh et al. [23], namely a DCGAN [20]. The generator produces a $64 \times 64 \times 3$ image using a randomly sampled variable from a 100 dimensional uniform distribution between $[-1, 1]$. The discriminator outputs a probability how likely a seen image is real. For training we use the Adam optimizer [14] with learning rate $\gamma = 0.002$, momentum $\beta_1 = 0.5$ and $\beta_2 = 0.9$ and batch size of 64 for 25 epochs.

During inpainting we use again the Adam optimizer to minimize the inpainting objective with respect to the latent variable z. We draw a random sample from the uniform distribution for initialization and restrict the values to $[-1, 1]$ along each dimension during optimization. We use the step size $\gamma = 0.003$, momentum $\beta_1 = 0.5$ and $\beta_2 = 0.9$. We use λ of 0.003 in all our experiments and use $\mu \in [0.2, 0.66]$ depending on the extent of the gap. We perform 1500 gradient descent steps during inpainting and refinement. We use TensorFlow [1] as deep learning framework on a Nvidia GeForce GTX TITAN X for training the GAN, inpainting and refinement. Refinement including inpainting takes approximately 4.7 s per image (1500 iterations each).

We used CVX, a package for specifying and solving convex programs [8, 9], to minimize the objective of the blending problem,. Minimization takes approximately 6 s per image in Matlab on a an Intel I7u CPU. We empirically observed highest quality results when using the Laplacian of Gaussians as discrete differentiation operator D, that we convolve with the corresponding image to approximate the gradient. The Laplacian of Gaussian is a 3×3 kernel with all 1 entries except the central element which is -9.

3.3 Mask or Hole Patterns

We use six different mask patterns: (1) and (2) a squared random positioned mask with 25% and 50% damaged or missing pixels, (3) a mask consisting of a vertical and horizontal stripe, (4) a mask where roughly 50% of the left hand side of the face is missing, (5) a centered mask where 25% of pixels are missing, (6) a mask where 70% of randomly selected pixels are missing. We use $\mu = 0.66$ for pattern (1) and (5), $\mu = 0.33$ for pattern (2) and (4) and $\mu = 0.2$ for pattern (3) and (6).

3.4 Results

We compute four classical metrics for image inpainting with and without refinement using the testing set. The metrics are Mean Absolute Error (MAE), Mean Squared Error (MSE), Peak Signal to Noise Ratio (PSNR) and Mean Structural Similarity Index Measure (MSSIM) [22]. Table 1 shows the scores of each metric before and after refinement. We report the better score in bold. We conclude that refinement improves always for each metric and all masks except the mask with randomly missing pixels. For this experiments the differences before and after refinement is beyond the reported precision such that the reported numbers agree. Note, that these metrics need a reference image such as the undamaged image. Nonetheless, a high quality inpainted region might look very different than the original one and can lead to a lower inpainting score. Therefore, we show more examples of inpainting with and without refinement in Figs. 9 and 10.

Table 1 Results of all testing images before and after refinement using six different hole patterns and four common metrics in image inpainting

	Random square (1)		Big random square (2)		Stripes (3)	
Refinement	Before	After	Before	After	Before	After
MAE ↓	16.20	**14.04**	35.60	**32.75**	39.43	**35.01**
MSE ↓	777.16	**626.17**	1852.22	**1599.89**	2011.45	**1607.32**
PSNR ↑	19.23	**20.16**	15.45	**16.09**	15.45	**16.07**
MSSIM ↑	0.8760	**0.8921**	0.7411	**0.7624**	0.707	**0.740**

	Left (4)		Center (5)		Random (6)	
Refinement	Before	After	Before	After	Before	After
MAE ↓	34.87	**31.62**	15.18	**13.68**	16.08	16.08
MSE ↓	1980.77	**1645.05**	704.54	**570.18**	323.97	323.97
PSNR ↑	15.16	**15.97**	19.65	**20.57**	23.03	23.03
MSSIM ↑	0.7601	**0.7823**	0.8716	**0.8840**	0.8908	0.8908

We highlight the superior performance (before or after) in bold. The arrows next to the metrics denote whether a lower score (↓) or a higher score (↑) is better. Note, that the difference of the random pattern between before and after is smaller than the displayed precision. Hence, we omit reporting any score in bold

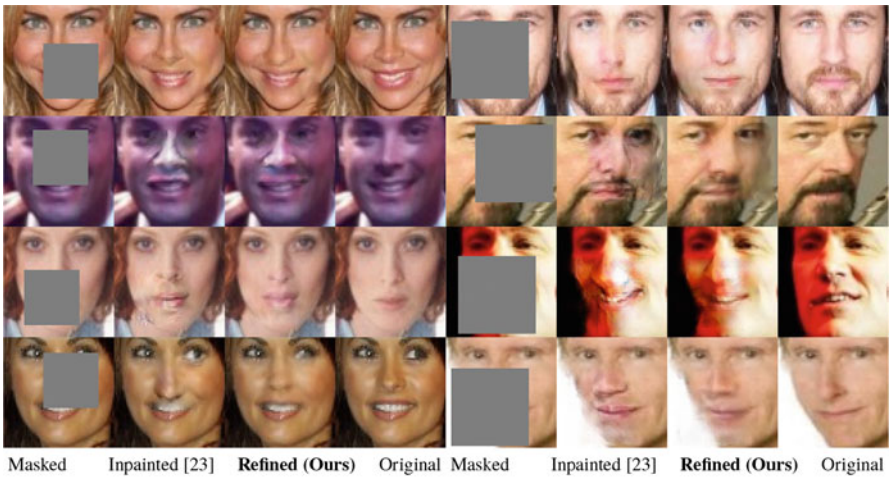

Fig. 9 Additional randomly selected images that highlight that the images after refinement are of a higher perceptional quality, corrected artifacts and look more plausible

To highlight that inpainting for 1500 steps followed by refinement for 1500 steps leads to a much higher quality than running inpainting for 3000 steps, we report the results for the centered hole pattern (5) in Table 2. See, Fig. 10 for sample images using 1500 or 3000 gradient steps for image inpainting. The quality of running inpainting for 3000 steps is marginally higher than using 1500 iterations. However, using refinement improves significantly for all metrics.

4 Discussion

Refinement improves in average for any hole pattern when considering different metrics. We show that refinement corrects visual artifacts caused by the locally limited content loss of classical semantic image inpainting. Furthermore, the synthetic image restored by the refinement GAN is much closer to the damaged image such that the perception score is more reliable and improves the inpainting. We observe, that refinement inpaints regions with similar face attributes than the damaged image, see Fig. 6. Although the restored images are often clearly visible distinct from the ground truth, most of the results are plausible and appealing. Nonetheless, we observed some failure cases, that we show in Fig. 11. We consider a refinement as failure if refinement reduces the image quality or fails to correct strong artifacts that were previously inpainted. Possible reasons are that the minimization approach did not converge fast enough, is stuck in an inappropriate local minimum or the initial inpainted solution has a very low quality, such that the inpainting loss prevents improving the reconstruction. We observed in Table 1, that refinement does not

Fig. 10 Additional examples showing the results for different hole patterns inpainted using 1500 or 3000 iterations with or without blending and the result of our refinement method with blending

Table 2 Comparison between running inpainting for 1500 or 3000 iterations or spending 3000 iterations for the proposed method (inpainting 1500 and refinement 1500)

	Inpainting		Inpainting and refinement
Iterations	1500	3000	1500 and 1500
MAE ↓	15.18	15.03	**13.68**
MSE ↓	704.54	694.53	**570.18**
PSNR ↑	19.65	19.71	**20.57**
MSSIM ↑	0.872	0.874	**0.8840**

We report the results using all testing images damaged with a centered square pattern.We highlight the superior performance (before or after) in bold. The arrows next to the metrics denote whether a lower score (↓) or a higher score (↑) is better

Fig. 11 Sample images where refinement either fails to correct the failures produced by semantic inpainting or where refinement degrades the image quality

improve the inpainting of an image where randomly selected pixels are damaged. The reason is, that the corresponding scores even without refinement are the highest among all other, that belong to different hole patterns. Hence, refinement is unable to further improve the inpainting because the difference between refinement and inpainting is much smaller for this hole pattern because almost every clean pixel is next to a hole and is therefore considered during inpainting. Similarly, the synthetic and the damaged image are much closer such that the perception loss is more reliable.

5 Conclusion

To summarize, in this paper we propose a new refinement GAN. We apply it on top of classical semantic image inpainting and design the corresponding pipeline. We tackle the challenging problem of reconstructing missing parts of human faces. We choose this problem and dataset because humans are especially sensitive even for tiny artifacts in faces. Therefore, plausible results for a human observer have already a very high quality. We show that refinement increases or achieves the same performance for any hole pattern for all measured quality metrics. However,

additional computations are required and the computational cost is twice as large. Nonetheless, refinement suppresses many disturbing failures of image inpainting such as inpainting moustaches in a female face, disagreeing eye or skin color or adding only a fraction of eyeglasses. Hence, we conclude that refinement is crucial for high quality image inpainting, where plausible and visually appealing results are desired and the additional computational cost is tolerable.

Acknowledgement This work was partly supported by ETH General Fund and a Nvidia GPU grant.

References

1. M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.
2. C. Barnes, E. Shechtman, A. Finkelstein, and D. B. Goldman. PatchMatch: A randomized correspondence algorithm for structural image editing. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 28(3), Aug. 2009.
3. A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40(1):120–145, May 2011.
4. T. F. Chan, J. Shen, and H.-M. Zhou. Total variation wavelet inpainting. *Journal of Mathematical Imaging and Vision*, 25(1):107–125, Jul 2006.
5. A. Criminisi, P. Perez, and K. Toyama. Object removal by exemplar-based inpainting. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages II–II, June 2003.
6. B. Dolhansky and C. Canton Ferrer. Eye in-painting with exemplar generative adversarial networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.
7. I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 2672–2680, 2014.
8. M. Grant and S. Boyd. Graph implementations for nonsmooth convex programs. In V. Blondel, S. Boyd, and H. Kimura, editors, *Recent Advances in Learning and Control*, Lecture Notes in Control and Information Sciences, pages 95–110. Springer-Verlag Limited, 2008.
9. M. Grant and S. Boyd. CVX: Matlab software for disciplined convex programming, version 2.1. http://cvxr.com/cvx, Mar. 2014.
10. C. Guillemot and O. L. Meur. Image inpainting: Overview and recent advances. *IEEE Signal Processing Magazine*, 31(1):127–144, Jan 2014.
11. S. Gutta, M. Trajkovic, A. J. Colmenarez, and V. Philomin. Method and apparatus for automatic face blurring, Oct. 25 2005. US Patent 6,959,099.
12. S. Ioffe, L. Williams, D. Strelow, A. Frome, and L. Vincent. Automatic face detection and identity masking in images, and applications thereof, Jan. 17 2012. US Patent 8,098,904.
13. P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

14. D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.
15. D. P. Kingma and M. Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.
16. Z. Liu, P. Luo, X. Wang, and X. Tang. Deep learning face attributes in the wild. In *The IEEE International Conference on Computer Vision (ICCV)*, 2015.
17. J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online dictionary learning for sparse coding. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pages 689–696, New York, NY, USA, 2009. ACM.
18. S. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin. An iterative regularization method for total variation-based image restoration. *Multiscale Modeling & Simulation*, 4(2):460–489, 2005.
19. D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros. Context encoders: Feature learning by inpainting. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2536–2544, 2016.
20. A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.
21. J. Shen and T. F. Chan. Mathematical models for local nontexture inpaintings. *SIAM Journal on Applied Mathematics*, 62(3):1019–1043, 2002.
22. Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.
23. R. A. Yeh, C. Chen, T. Y. Lim, A. G. Schwing, M. Hasegawa-Johnson, and M. N. Do. Semantic image inpainting with deep generative models. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5485–5493, 2017.
24. M. Zhou, H. Chen, L. Ren, G. Sapiro, L. Carin, and J. W. Paisley. Non-parametric bayesian dictionary learning for sparse image representations. In Y. Bengio, D. Schuurmans, J. D. Lafferty, C. K. I. Williams, and A. Culotta, editors, *Advances in Neural Information Processing Systems 22*, pages 2295–2303. Curran Associates, Inc., 2009.