Image Inpainting with Conditional Diffusion Model

- Course Project Report

Anandu A S
Roll No :213074001
EE1, MTech (2021-23)
Second year, EE Dept, IITB,

213074001@iitb.ac.in

Dharshan Sampath Kumar
Roll No :18d180009
CMinds, Dual Degree (2018-23)
IITB,

18d180009@iitb.ac.in

I. Introduction

Image inpainting is the process of recreating missing or damaged portions of an image. The challenge in image inpainting is to produce visually plausible structure and texture for the missing regions of the image. Diffusion model in deep learning was first introduced by Sohl Dickstein et al in the paper "Deep Unsupervised Learning using Non equilibrium Thermodynamics" in 2005. Diffusion model is a generative latent variable model. They define a Markov chain of diffusion steps to slowly add random noise to data and then learn to reverse the diffusion process to construct desired data samples from the noise. The model learns the distribution of the training samples and the model can generate new image samples by sampling from this distribution. The paper "ILVR: Conditioning Method for Denoising Diffusion Probabilistic Models" by Jooyoung by et al in ICCV 2021 tries to condition the generative process by adding reference images to the reverse diffusion pipeline. Our objective is to use intermediate latent variable refine (ILVR) model by Jooyoung et al for inpainting task. The challenge in this problem is that as images are sampled from distribution, the edges and corners may not exactly overlap in the masked portion for inpainting task. The challenge is to recreate the image as close to the original image.

II. Diffusion Models

Diffusion model define a Markov chain of diffusion steps to slowly add random noise to data and then learn to reverse the diffusion process to construct desired data samples from the noise. The forward process adds small amount of Gaussian noise to the sample in T steps producing a noisy sample sequence $\mathbf{x}_1, \dots, \mathbf{x}_T$. The step sizes are controlled by a variance schedule $\{\beta_t \in (0,1)\}_{t=1}^T$.

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I}) \quad q(\mathbf{x}_{1:T}|\mathbf{x}_0) = \prod_{t=1}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1})$$

As T becomes very large, the images acquires the nature of an isotropic Gaussian distribution.

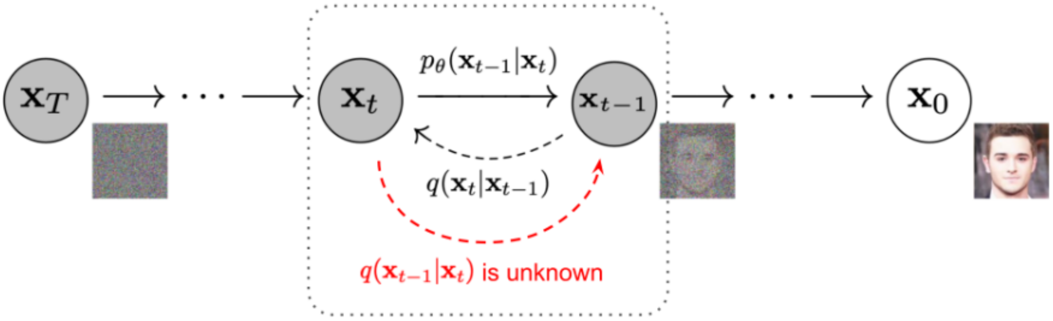

$\mathbf{x}_T$
$\mathbf{x}_t$
$p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)$
$\mathbf{x}_{t-1}$
$q(\mathbf{x}_t|\mathbf{x}_{t-1})$
$q(\mathbf{x}_{t-1}|\mathbf{x}_t)$ is unknown
$\mathbf{x}_0$

Fig 1 (Diffusion model forward and reverse process)

The sample xt at any arbitrary time step t can be computed in a close form using reparameterization trick.

$$\begin{aligned}
\mathbf{x}_t &= \sqrt{\alpha_t}\mathbf{x}_{t-1} + \sqrt{1-\alpha_t}\boldsymbol{\epsilon}_{t-1} && \text{;where } \boldsymbol{\epsilon}_{t-1}, \boldsymbol{\epsilon}_{t-2}, \dots \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \\
&= \sqrt{\alpha_t\alpha_{t-1}}\mathbf{x}_{t-2} + \sqrt{1-\alpha_t\alpha_{t-1}}\bar{\boldsymbol{\epsilon}}_{t-2} && \text{;where } \bar{\boldsymbol{\epsilon}}_{t-2} \text{ merges two Gaussians (*).} \\
&= \dots \\
&= \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon} \\
q(\mathbf{x}_t|\mathbf{x}_0) &= \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1-\bar{\alpha}_t)\mathbf{I})
\end{aligned}$$

During training, the model learns to reverse this diffusion process in order to generate new data. Starting with pure Gaussian noise $p(\mathbf{x}T) := \mathcal{N}(\mathbf{x}T, \mathbf{0}, \mathbf{I})$, the model learns the joint distribution $p\theta(\mathbf{x}0{:}T)$

$$p_\theta(\mathbf{x}_{0:T}) := p(\mathbf{x}_T)\prod_{t=1}^{T} p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) := p(\mathbf{x}_T)\prod_{t=1}^{T} \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \boldsymbol{\Sigma}_\theta(\mathbf{x}_t, t))$$

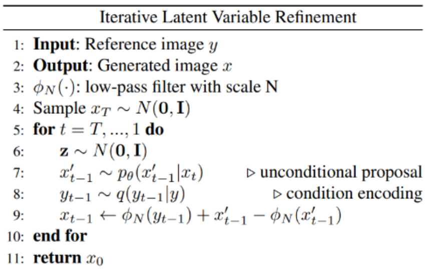
III. Intermediate latent variable refinement diffusion models

The paper by Jooyoung et al tries to control the reverse diffusion process by adding a reference image in the reverse pipeline. The images are sample in the reverse process from the conditional distribution $p(x_0|c)$ with the condition c .

$$p_\theta(x_0|c) = \int p_\theta(x_{0:T}|c)dx_{1:T},$$

$$p_\theta(x_{0:T}|c) = p(x_T)\prod_{t=1}^{T} p_\theta(x_{t-1}|x_t, c).$$

Each transition $p_\theta(x_{t-1}|x_t, c)$ of the generative process depends on the condition c. In order to avoid hard conditioning, a sequence of down sampling and up sampling ($\phi_N(y)$)by a factor of N is added in the reverse pipeline. As the value of N increases, the distance of the generated image from the reference image increases and vice versa.



```
Iterative Latent Variable Refinement
Input: Reference image $y$
Output: Generated image $x$
$\phi_N(\cdot)$: low-pass filter with scale N
Sample $x_T \sim N(\mathbf{0}, \mathbf{I})$
for $t = T, ..., 1$ do
    $\mathbf{z} \sim N(\mathbf{0}, \mathbf{I})$
    $x'_{t-1} \sim p_\theta(x'_{t-1}|x_t)$        ▷ unconditional proposal
    $y_{t-1} \sim q(y_{t-1}|y)$        ▷ condition encoding
    $x_{t-1} \leftarrow \phi_N(y_{t-1}) + x'_{t-1} - \phi_N(x'_{t-1})$
end for
return $x_0$
```

Table1 (Intermediate Latent Variable Refinement Algorithm)

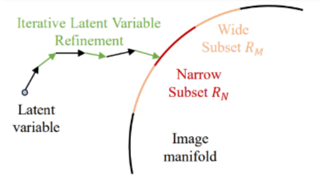


Fig 2 (Intermediate Latent Variable Refinement process flow)

IV. Image Inpainting

Image inpainting is the class of techniques used for restoration methods used to remove damaged or unwanted objects from an image. The challenge in image inpainting is to refill the damaged areas of the image preserving the semantic context of the overall image. In mask blind inpainting, the model automatically detects damaged areas (or black mask areas) in the input image and fill those areas. In mask known inpainting, the damaged or missing portions in image is given as an additional mask. The model makes use of this mask information to recreate the damaged or missing portions in the image. Mask agnostic models are trained to work with different shapes of masks and recreate images irrespective of the shape of the mask.

V. Methods

The following approaches were carried out as part of the project.

Blind mask inpainting without separate mask prediction

In blind mask inpainting, mask information is not provided with the image. In this approach, we have taken the Hadamard product of the masked area to the reference image. This output was given to the model. Separate mask information was not given to the image. This image is given as conditional image to the ILVR diffusion model. From the recreated image, the masked area (area to be recreated) was cropped out and overlayed on top of the input conditional image to create the final output image. Multiple experiments were carried out by changing the hyperparameters like no of diffusion steps, conditioning steps, resampling steps and the upsampling / downsampling factor for intermediate latent variable refinement.

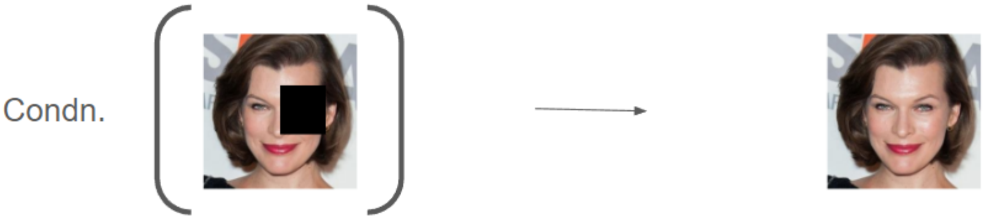


Fig 3 (Blind mask inpainting with ILVR)

We have noticed that either

1. The masked portion in the recreated image is blank or
2. The recreated image is totally different from the reference image which totally destroys the objective of image inpainting task.

   From these observations, we have come to the conclusion that the backward diffusion process instead of recreating from the conditional image, it merely recreates the blank(black) portion by removing the noise which was added to it during the forward noising process. So the features of the mask (masked area) have to be explicitly replaced by an isotropic gaussian noise reverse diffusion intermediate steps so as to remove its effect in the reverse diffusion process. We believe that the results that we observed are due to the incomplete passage of information resulting from solely relying on the conditioning image and not incorporating the generated image. Therefore we propose combining the unmasked portion of the image (from forward diffusion step) and masked portion of image from reverse diffusion step (starting from isotropic gaussian noise) in order to improve the results.

Known mask inpainting

In this method, the mask is given as a separate input to the model. In this approach, we have taken the Hadamard product of the masked area to the **noised** conditioning image. This is used as conditional image for the unmasked portions of the image. For the masked portions of the image, we started from a complete isotropic gaussian noise and did the reverse diffusion process. The masked area of reverse diffusion image at T^{th} step is concatenated with the unmasked area of the reference image at the same step (forward process) generated using the formula $\mathbf{x}_t = \sqrt{\alpha_t}\mathbf{x}_{t-1} + \sqrt{1-\alpha_t}\boldsymbol{\epsilon}_{t-1}$. This process enables mixing of information from conditional as well as generated image at each step of diffusion before passing it to the successive iteration. Therefore, the approach brings the distribution closer between masked and unmasked portions of the image and thus creates a continues image that is congruent at the edges of the original mask.

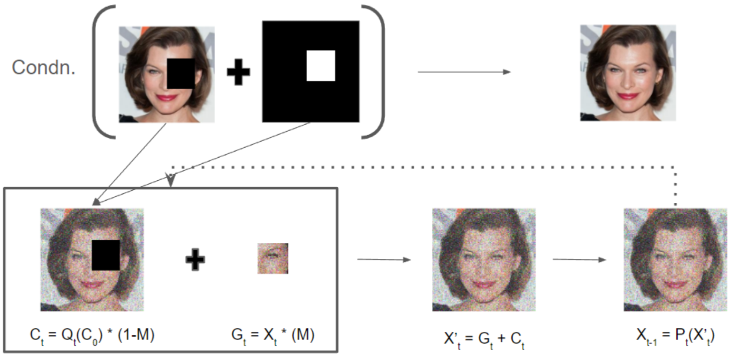


Fig 4 (Known mask inpainting with ILVR)

Blind mask inpainting with separate mask prediction

In this method, we have trained a separate network based on UNet to predict the masked portion in the image. The reference image with the blind mask (generated by the mask prediction model) is given as input to the algorithm discussed above to generate the inpainting output even when the mask information is not explicitly available.

Using image inpainting for super resolution.

In this method, we have tried to used ILVR diffusion process for image super resolution task(2x). We have tried 3 approaches for this task.

1. We have bicubic interpolated the original low-resolution image by a factor of 2. Then we created a mask with each alternate row and column as blank space. We then used this mask and the high resolution image as reference using the method discussed above (Known mask inpainting) to create the final high resolution output.
2. In this approach, we have upsampled the original image by just introducing blank column and row alternatively. Mask was created with alternate row and column as blank spaces. We then used this mask and the upsampled image as reference using the method discussed above (Known mask inpainting) to create the final high-resolution output.

3. In this method, we up sampled the image using bicubic interpolation. We then fed this image as reference image to ILVR diffusion model to generate high quality output.

Using auto regressive approach for image inpainting

In this method, we tried to generate high quality images with autoregressive image inpainting. We have taken LaMa (Large mask inpainting) model which is trained for image inpainting. The approach was to see whether autoregression can improve the image inpainting performance. Towards this, we have continuously eroded the mask size by 1 pixel width and generated new image inpainting samples. The idea was that, given the context of additional pixel, whether the image inpainting model can predict better result than predicting all pixel outputs in a single go.

VI. Results And Conclusions

Blind mask inpainting without separate mask prediction

Test 1

Diffusion steps : 1000
Timespace respacing :1000
Range_t :100
Upsampling/downsampling factor : 1

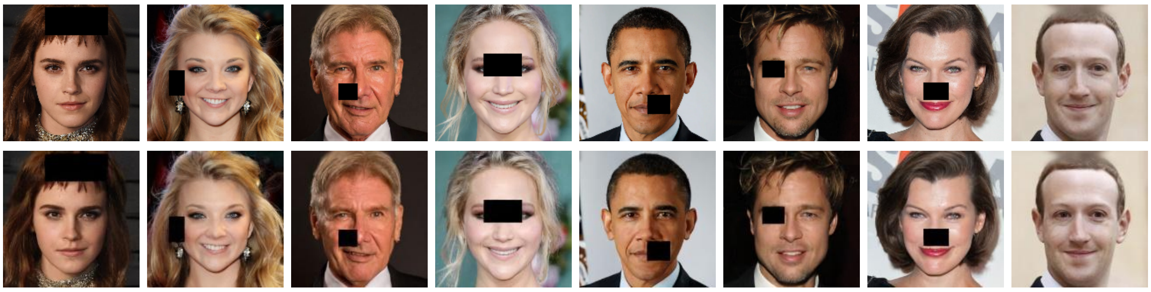


Fig 5 (Reference image given in first row and inpainted image shown in second row)

Test 2

Diffusion steps : 1000
Timespace respacing :1000
Range_t :60
Upsampling/downsampling factor : 1

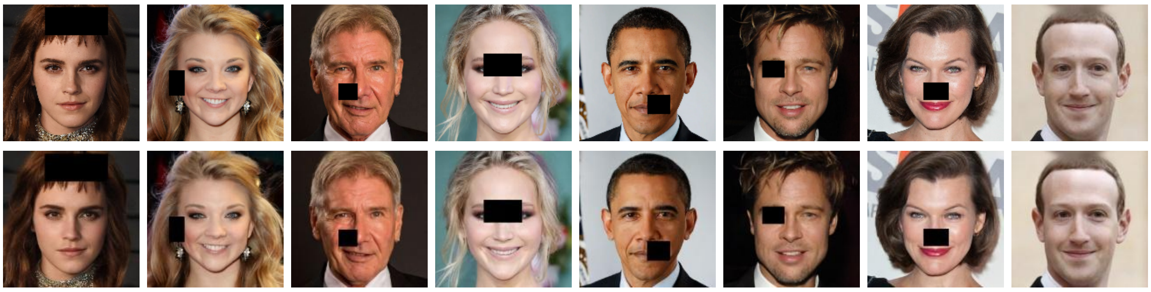


Fig 6 (Reference image given in first row and inpainted image shown in second row)

Test 3

Diffusion steps : 1000
Timespace respacing :1000
Range_t :60
Upsampling/downsampling factor : 2

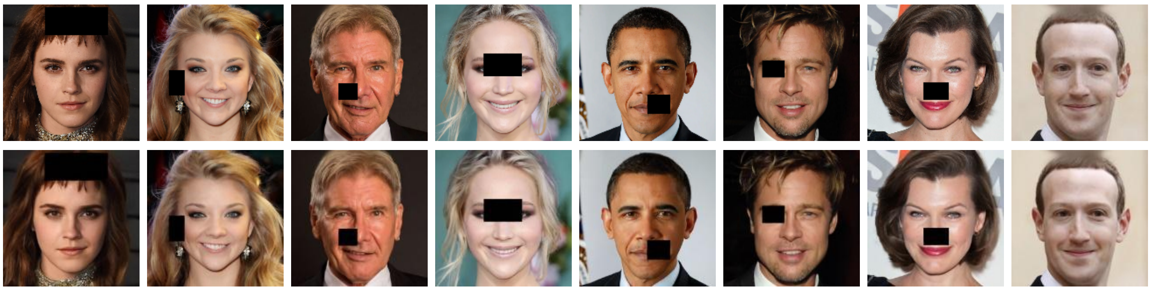


Fig 7 (Reference image given in first row and inpainted image shown in second row)

Test 4
Diffusion steps : 1000
Timespace respacing :1000
Range_t :60
Upsampling/downsampling factor : 16

Fig 8 (Reference image given in first row and inpainted image shown in second row)

Test 5
Diffusion steps : 1000
Timespace respacing :110
Range_t :60
Upsampling/downsampling factor : 1

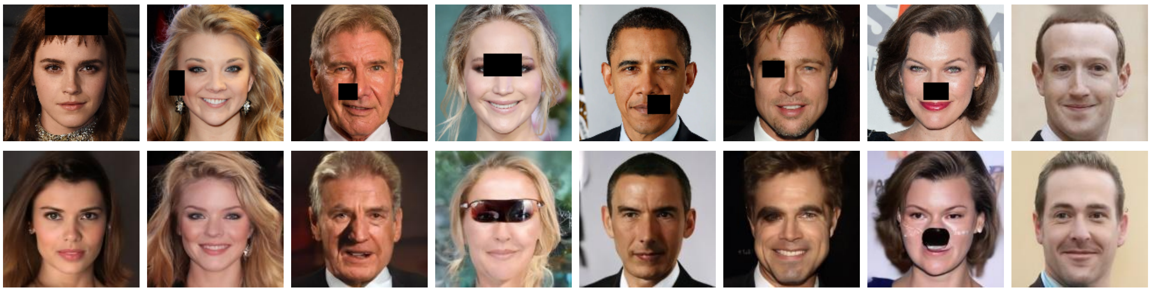


Fig 9 (Reference image given in first row and inpainted image shown in second row)

Test 6
Diffusion steps : 1000
Timespace respacing :100
Range_t :100
Upsampling/downsampling factor : 1

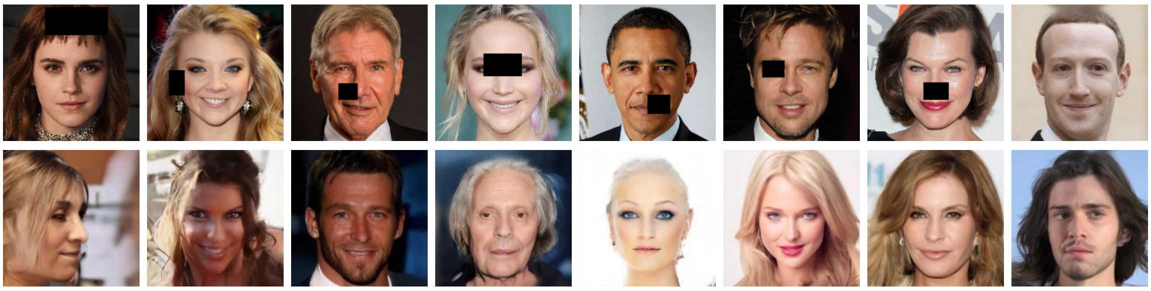


Fig 10 (Reference image given in first row and inpainted image shown in second row)

The conclusion from these experiments is this approach is not suitable for image impainting because

1. Either mask portions are retained in image
2. Or the image is changed fully which will beat the purpose of image inpainting.

	Blind	Known Mask (small)	Known Mask (Large)
LPIPS	0.175	0.035	0.115
SSIM	0.802	0.965	0.843

Known mask inpainting

Test 1

Diffusion steps : 1000
Timespace respacing :1000
Range_t :100
Upsampling/downsampling factor : 1

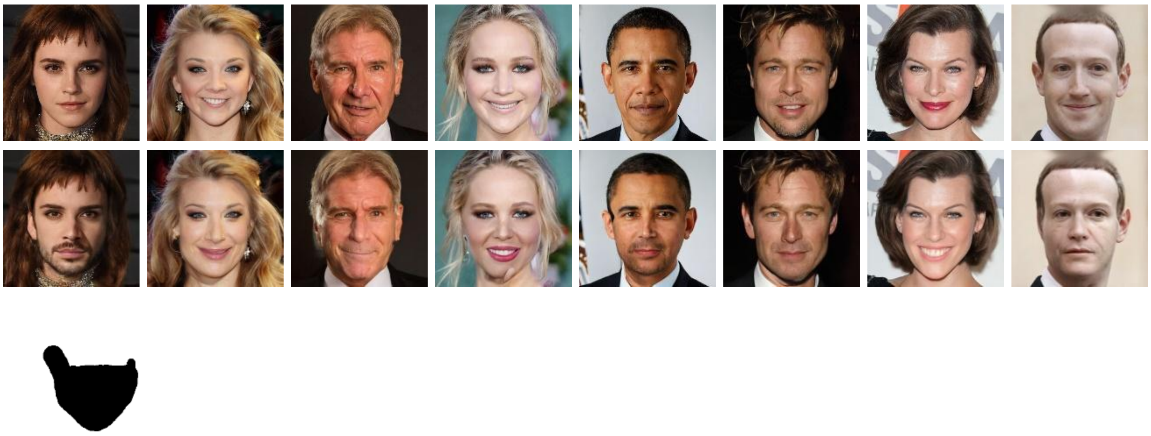


Fig 11 (Reference image given in first row, inpainted image shown in second row and mask is given in third row)

Test 2

Diffusion steps : 1000
Timespace respacing :1000
Range_t :100
Upsampling/downsampling factor : 1

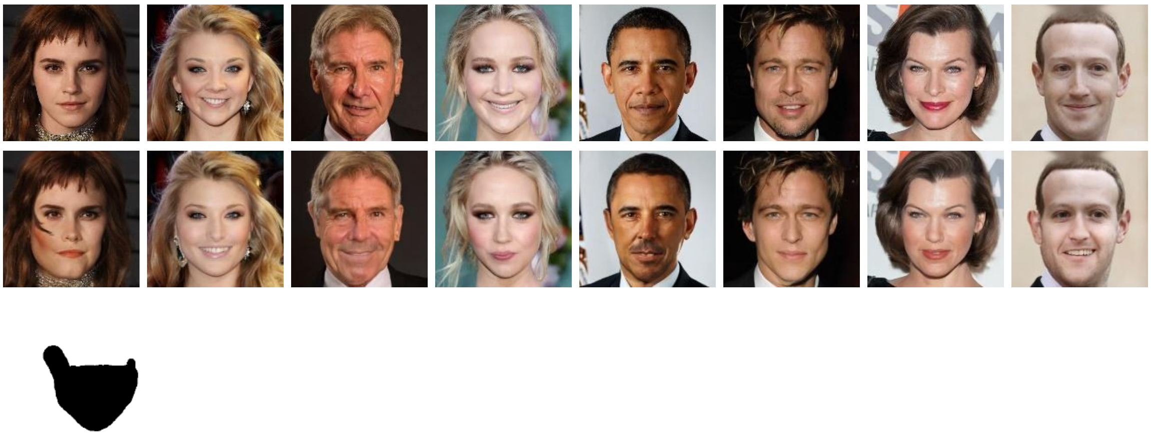


Fig 12 (Reference image given in first row, inpainted image shown in second row and mask is given in third row)

Test 3: Inpainting with large mask

Diffusion steps : 1000
Timespace respacing :1000
Range_t :100
Upsampling/downsampling factor : 1

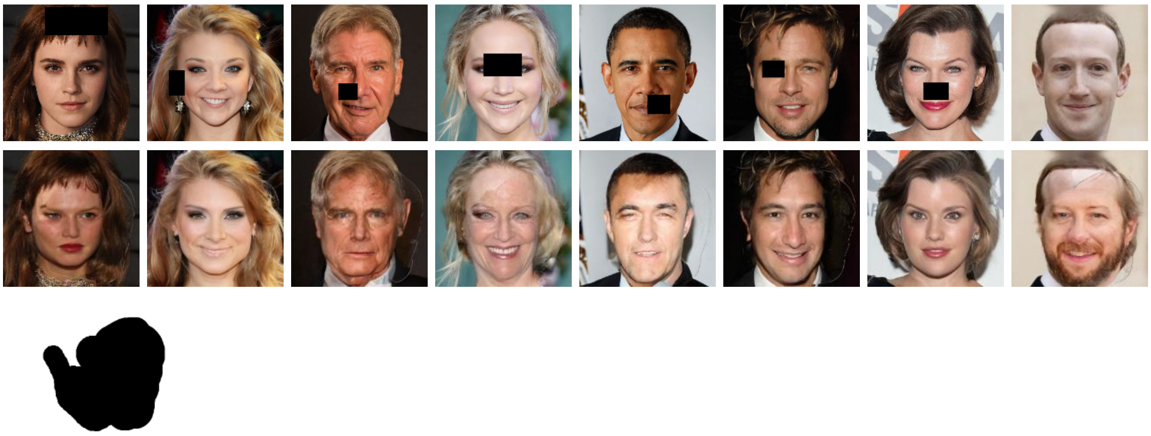


Fig 13 (Mask in first row ,Reference image given in second row and inpainted image shown in second row)

The conclusion from these experiments are as follows

1. For small masks, the generated results are good for inpainting task. In the reverse diffusion process, the starting point is an arbitrary point in the latent variable space. However the addition of conditional image information forces the model to move to the mode of conditioning image. Thus generated results are very much identical to the conditional image (in the unmasked region).
2. However as mask size is increased, we have observed mode collapse problem. This is because in the reverse diffusion process, the starting point is an arbitrary point in the latent variable space. The amount of information in the conditioning image is small enough to pull the whole output to the mode of the conditioning image. Thus the final output image fails to reach the mode of the conditioning image and this results in discontinuity/texture mismatch or artefacts in the output image.

	Bi cubic cond	As Inpainting	Bicubic
PSNR	28.331	22.640	33.747
SSIM	0.984	0.943	0.994

Using image inpainting for super resolution.

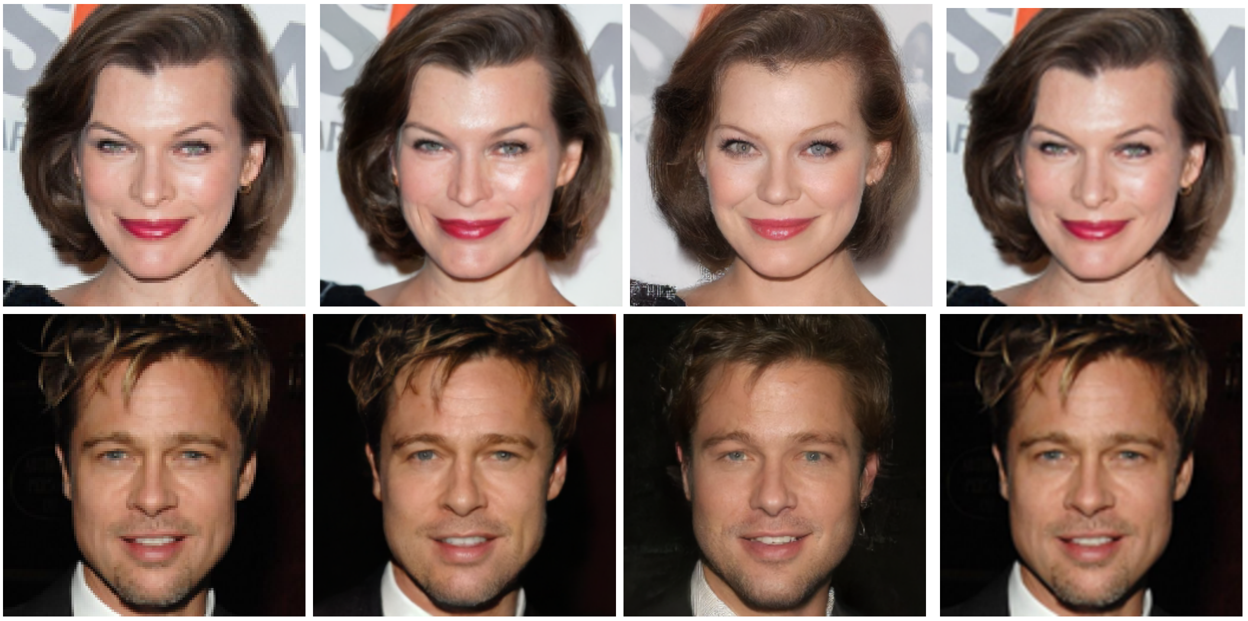


Fig 13 (Col 1: Low Resolution img(128*128), Col 2: Bicubic followed by ILVR, Col 3: Bicubic followed by masked ILVR diffusion, Col 4: Bicubic interpolation)

The conclusion from these experiments are as follows.

1. On visual inspection the quality of images is as follows (increasing order)
   a. Bicubic followed by Masked ILVR diffusion (258*26)
   b. Bicublic followed by ILVR (256*256)
   c. Bicubic interpolation (256*256)
   d. Low resolution image (128*128)
2. However on performance metrics like PSNR and SSIM, the order is as follows (increasing order)
   a. Bicubic interpolation (256*256)
   b. Bicublic followed by ILVR (256*256)
   c. Bicubic followed by Masked ILVR diffusion (128*128)

   The possible interpretation of this anomaly is that diffusion operation may slightly change the image from the original input to (as it is a sampling process) which can decrease performance metrics like SSIM and PSNR even when the visual quality of the image is improved.

VII. References

1. *Denoising Diffusion Probabilistic Models by Jonathan Ho, Ajay Jain, Pieter Abbeel*
2. *ILVR: Conditioning Method for Denoising Diffusion Probabilistic Models by Jooyoung Choi, Sungwon Kim, Yonghyun Jeong, Youngjune Gwon, Sungroh Yoon*
3. *LaMa: Resolution-robust Large Mask Inpainting with Fourier Convolutions by Roman Suvorov, Elizaveta Logacheva, Anton Mashikhin, Anastasia Remizova, Arsenii Ashukha, Aleksei Silvestrov, Naejin Kong, Harshith Goka, Kiwoong Park, Victor Lempitsky.*
4. *Github repository : https://github.com/remag2069/CS726_project*