Image Super-Sampling and Reconstruction from Sparse Samples

Yiming Dou, Qing Yang

Abstract:

Although real-time rendering is emerging, high processing latency, high resource demand and specific hardware requirement hinder its further employment. To explore the possibility of rendering image with high performance but low latency and demand, Sparse sampling methods that sample the origin image into sparse grids or patches are considered. In this project, two sampling methods are proposed and implemented, and they are defined as **Sparse-Grid Sampling** and **Sparse-Patch Sampling**, which respectively sample the origin images into sparse grids or sparse patches. Specifically, the sampling process is based on FFT, which is shown in this project to bring excellent down-sampling effect. For **each** sparse sampling algorithm, a method is designed and implemented in this project in order to obtain the HR origin images based on the sparse samples. For Sparse-Grid Sampling, the Single Image Super-Resolution (SISR) is performed to obtain the HR images. **Three different networks: SRCNN, DRRN and UNet** are tested in this project, and DRRN largely outperforms other methods when it comes to the PSNR results. In Sparse-Patch Sampling scenario, the newly-proposed **Masked Auto Encoder (MAE)** is used to reconstruct the origin image from the sparse sample patches, and it also has great effect for image sparse-patch sampling reconstruction. The code has been made available to public: https://github.com/Dou-Yiming/CS337-Project

Key word：Voronoi-Delaunay Triangulation, FFT, Single Image Super Resolution, Masked-Auto-Encoder

1 Introduction

1.1 Significance

Real-time rendering application is ubiquitous in modern media such as mobile phone and virtual reality. With increasing display and resolution demands, high processing latency and high resources used pose a challenge for today's rendering philosophy. Fixed foveated rendering renders peripheral regions at low resolutions. Microsoft reduces shading complexity for foveated and high-resolution displays using variable rate shading. However, these methods rely heavily on specific hardware; or brings some artificial features in image which lower the display quality.

In this paper, two methods of sparse-sampling are proposed, including a sampling algorithm based on Delaunay Triangulation and a patch-sampling algorithm based on FFT high-pass filter. After the sampling, the image is stored as sparse grids instead of traditional dense matrix. In order to reconstruct the original image from the sparse grids, two image reconstruction algorithms are designed accordingly, including image super-sampling based on deep learning that uses SRCNN, DRRN or UNet and a image reconstruction algorithm based on Masked Auto Encoder (MAE). To summarize, the contribution of this article is threefold:

1. Introducing a novel way of storing and representing images as sparse grids instead of dense grids
2. Proposing a sparse-grid sampling method based on FFT and obtain LR images by Delaunay Triangulation and accordingly design a model to reconstruct the original image based on the sparse grids.
3. Proposing a sparse-patch sampling method based on image patch splitting and FFT and design a model to reconstruct the original image based on the chosen image patches.

1.2 Article Structure

As is shown in the following figure, the article architecture consists of **two independent paths** that respectively proposes an image sampling and reconstruction system. In the first path, a sparse-grid sampling method is designed

upon Delaunay Triangulation and a model based on SRCNN, UNet or DRRN is designed to reconstruct the original image. In the second path, a sparse-patch sampling method is designed using FFT and patch splitting, while the image reconstruction algorithm is accordingly designed with the help of Masked Auto Encoder (MAE).

Each of the paths introduce a novel algorithm that sample the original image to sparse grids or patches and accordingly design an image super-sampling or reconstruction algorithm.

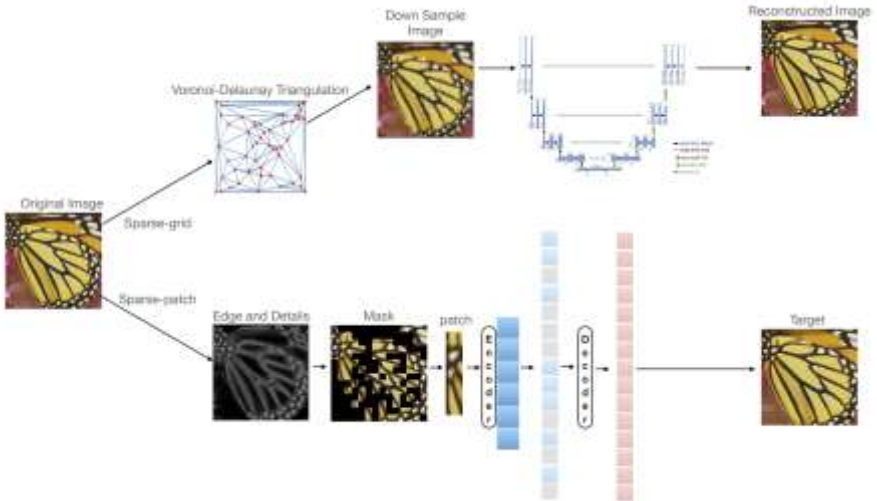

Fig 1: Workflow of this article, two methods are proposed and implemented

2 Related Works

2.1 Image Supersampling

Rendering the image at a much higher rsolution that the one being displayed and then calculating an average color value, supersampling removes aliasing to make realistic objects. Algorithms include Grid algorithm in uniform distribution, Rotated grid algorithm, random algorithm, Jitter algorithm, Posson disc algorithm, Quasi-Monte Carlo method algorithm, N-books, RGSS, HRAA, Filpquad, Fliptri have been proposed to against alias. However, basic supersamling methods are computational expensive due to the demand for video card memory and memory bandwidth. To solve this problem, machine learning or so called deep learning has been introduced to this field. Nvidia proposed deep learned supersamling(DLSS), exploiting the neural network to upsampling low resolution content. [28] 4x4 supersamling with high spatial and temporal fidelity has been achieved using temporal information and representation learning framework [29]. In our work, we also incorporate machine learning into the supersampling technique to reach a better display quality and lower computation cost.

2.2 Real-time Rendering

Real-time rendering focuses on analyzing and producing images in real time. It's widely used in today's computer, mobile, virtual reality and so on. Users can interact with the render as it is developed. Besides, real-time software rendering facilitate its application. However, demands for real-time requires large amount of computation resources; and specific hardware in some application.

2.3 Fast Fourier Transform

Fast Fourier transform (FFT) is an algorithm that widely implemented in programming language package and

compatible with GPU. FFT sampling can convert signal including the image to the spectrum domain. It's widely used in designing digital filter and fast processing of image because it can expose image features, like periodic interferences which is not visible in spatial domain and compress image into a more compact representation.

[26] evaluates the performances of FFT in real-time application and suitability for GPU implementation. Besides, conditions that FFT gives better performance have been identified. FFT is utilized to extract a sparse and collaborative for image classification [27]. In our discussion, FFT is used as a sampling method that lower the resolution of the original image thus the resource needed for following tasks.

2.4 Disentangled Representation Learning

Disentangled representation is desired as it represents a human interpretable pattern [1,2,3], enabling the downstream tasks learned more easily [4] and generalizes better [5]. In this paper, we try to use the disentangled property of represesntation learning to extract a disentangled latent space which is used as the input for the reconstruction.

We notice the recent study of disentanglement is promoted by two communities: Disentanglement in Deep Features and Independent Component Analysis. Their research previously lie on different assumptions, data patterns, and evaluation metrics.

One community is motivated by the newly raised deep learning for encouraging disentangled representation over independent factors. they have shown much empirical progress on this problem and they directly term their goal as ``disentanglement''. The related study is usually based on deep generative models. For instance, VAE-based methods have achieved successes on this task [6,7,8,9]. Besides, Generative Adversarial Networks (GAN) [2,10] are also put into the discussion of encouraging representations' disentanglement. More recently, people have shown that the GAN-based approach can achieve competitive performance as the above VAE variants [11,12,13]. A recent work [14] summarizes the popular methods and metrics in this community and proposes a tool for evaluation called disentanglement_lib, including popular metrics such as DCI [15], SAP [16], MIG [8] and so on. We use the encoder with verified high disentanglement score in the sampling.

Besides this series of studies, exploring underlying factors of variation in data pattern is a long-standing goal of the Independent Component Analysis (ICA) community [17]. They share many similarities, for example, generative models, e.g., VAEs, are recently popular in both [18,19]. ICA usually has different assumptions with the ``purely unsupervised learning'' [20]. For example, the pattern of noise [21,22] or some additional auxiliary variables [23,24] can be observed. Traditionally, ICA uses identifiability to assess their desired representation pattern and the popular metric is Mean Correlation Coefficient (MCC). SlowVAE [25] recently makes a great effort to connect the two branches of study but it still requires additional information such as temporal transition pattern.

3 Contents and Methods

In this article, two sparse-grid down-sampling methods are proposed, and two methods that reconstruct the origin image from sparse grids are designed accordingly.

3.1 Sparse-Grid Down-sampling

Traditional down-sampling methods regard the image as a matrix with several channels, and the sampling points

are covered on the original image densely and uniformly, which results in the low-resolution (LR) image being dense-and-uniform grids. This kind of methods may lead to several problems:

1. Data Redundancy: The position of sampling points are uniformly and densely generated, thus not being able to distinguish the detailed features in the image. Specifically, it is not reasonable to sample the areas with more details and those with less details from the same sampling-rate, since this may cause data-redundancy. If the same amount of data is used, and the data sampled from areas with more detail occupies a greater proportion, then more details of the image are retained, resulting in better effect.
2. Computation Redundancy: In addition to data redundancy, the computation redundancy is also a serious problem. Consider an image rendered from a model, much computation such as lighting and shading is needed, and these computations usually result in heavy cost. If the shading and lighting procedures of two sample points are similar with each other, then there is no need to sample both of them.

In order to solve these possible problems, two algorithms are proposed in this article, which use sparse sample points to down-sample the image.

3.1.1 Voronoi-Delaunay Triangulation Algorithm

In mathematics and computational geometry, a Voronoi-Delaunay Triangulation can be depicted as: Given a set P of discrete points in a general position, DT(P) is a triangulation such that no point in P is inside the circumcircle of any triangle in DT(P). Delaunay Triangulation maximize the minimum angle of all the angles of the triangles in DT(P).

In order to efficiently compute the triangulation process, the Divide-and-Conquer method is used, which has been shown to be the fastest method to compute Delaunay Triangulation. In this algorithm, a line that split the vertices into two non-overlap subsets is drawn recursively. Next, the Delaunay triangulation of each subset is computed, after which the two subsets are merged along the splitting line. The merge operation can be done within time O(n), thus the total running time is O(nlogn), which is acceptable.

In this article, two sampling algorithms are designed and implemented:

1. Random Sampling

   Given the number of sample points, the position of the points are randomly generated within the range of $(0,1)^2$. Then, the points are scaled to fit the size of the image and covered on the image, which is shown in the following figure.

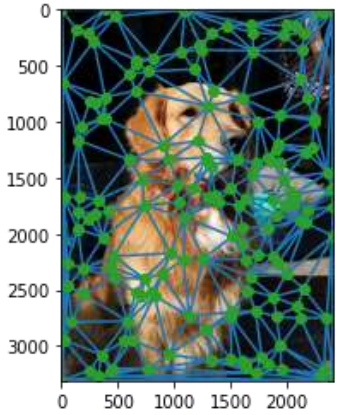

Fig 2: Sample points generated by random sampling algorithm

2. FFT-based Sampling

   Apparently, the random sampling algorithm has a drawback: it is not able to distinguish the areas with more

detailed features from those with less detailed features. As a result, this method may sample unnecessary points in the areas with relatively simple feature. (e.g. the left-up corner of Fig 2)

Therefore, to address this drawback, FFT high-pass filtering is utilized to extract the areas with higher signal frequency, which is shown in the following figure.

Fig 3: The result of FFT high-pass filtering

Areas with more detailed information such as the fur of dog are paid attention to

Now that we have obtained the areas with more details, higher sampling rate should be applied to those areas. The sampling process can be depicted as:

$$\bar{p} = \frac{\mathrm{N}_{\mathrm{samplepoints}}}{\sum_{\mathrm{eachpixel}} V}$$

$$P[i][j] = \bar{p} \times V[i][j]$$

In which $V[i][j]$ is defined as the value of position (i, j) of the FFT high-pass filtering image and $P[i][j]$ is defined as the sampling probability of position (i, j).

The method makes sure that the expected number of actual sampling points is equal to the set value.

Finally, the points are sampled based on the probability, and the result is shown in the following figure.

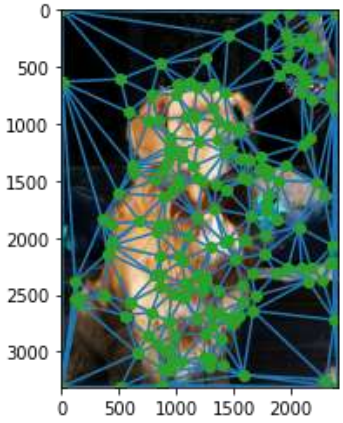

Fig 4: Sample points generated by FFT-based sampling algorithm

Clearly, the regions with more details are sampled with much higher sampling rate, which results in retaining relatively more information of the origin image with the same number of sampling points.

Next, the Delaunay Triangulation algorithm is performed, which generates triangles from the randomly-generated sample points. This results in each pixel's staying in one and only one triangle, making it possible to set the color of

each pixel based on the colors of the vertices of the triangle.

Specifically, the color of each pixel of the LR image is interpolated by the color of each vertex of the triangle, using the barycentric coordinate.

The process of computing the barycentric coordinates can be regarded as solving the following equations:

$$\begin{cases} P_x = iA_x + jB_x + kC_x \\ P_y = iA_y - jB_y + kC_y \\ i - j + k - 1 \end{cases}$$

The solution of the equations is:

$$\begin{cases} j = \dfrac{-(P_x - C_x)(A_y - C_y) + (P_y - C_y)(A_x - C_x)}{-(B_x - C_x)(A_y - C_y) + (B_y - C_y)(A_x - C_x)} \\ i = \dfrac{-(P_x - B_x)(C_y - B_y) + (P_y - B_y)(C_x - B_x)}{-(A_x - B_x)(C_y - B_y) + (A_y - B_y)(C_x - B_x)} \\ k = 1 - i - j \end{cases}$$

The comparison between the two sampling methods under 3, 100 and 1000 down-sampling rate is shown in the following figure. It is clear that when it comes to the detailed information, the FFT-based sampling algorithm largely outperforms the original random sampling method. The detailed comparison will be shown in the experiment section.

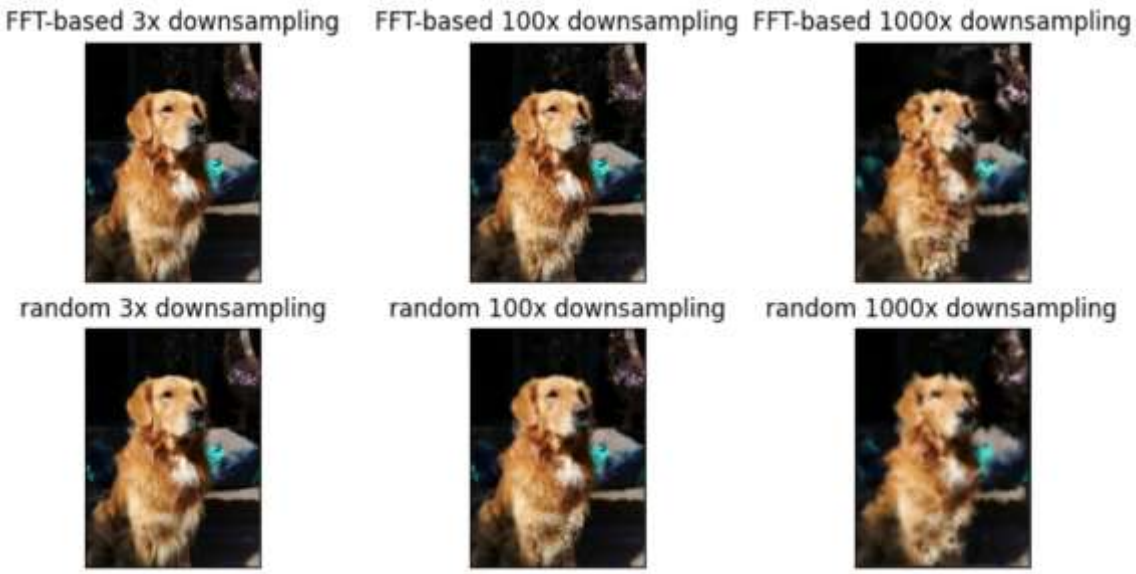

Fig 5: LR image generated by Delaunay Triangulation

3.1.2 Image Patch Splitting

This method follows the procedure of Vision Transformer (ViT). Each image is split into fixed-size patches. In the next step, the patches that contain more detailed information should be sampled, since these patches bring more information into the model and tend to help it reconstruct the original image better.

To find out these patches, a high-pass filter based on FFT is used to extract the texture and edge features of the image. The process can be depicted in the following figure:

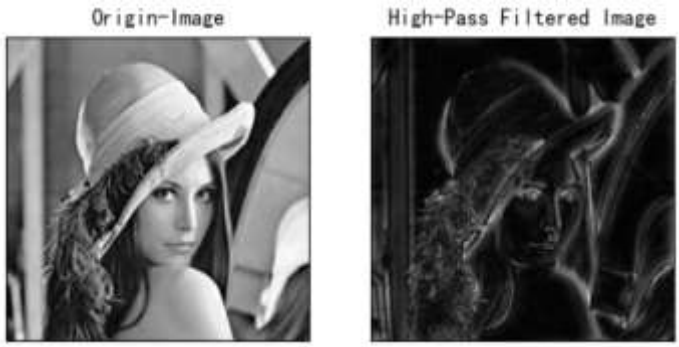

Fig 6: High-pass filtering based on FFT

After the high-pass filtering, the areas with more details are retained, making it possible to select the patches that provides more information. Specifically, to show the effect of selecting patches based on FFT high-pass filtering, we adopt two selection strategies: (1) Random-Selection, (2) FFT-based-Selection. The result of each strategy is shown in the following figure (the patches that are not selected are depicted as masks):

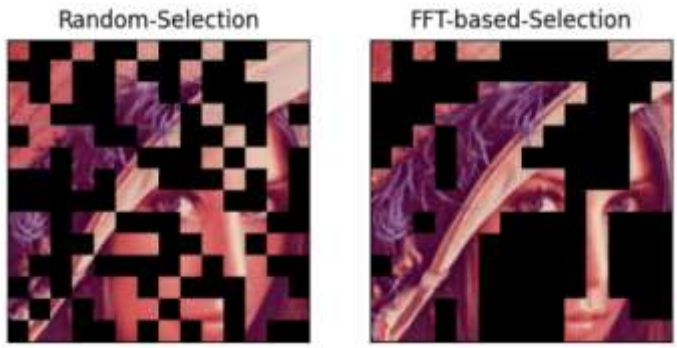

Fig 7: Comparison between two selection strategies

Therefore, the FFT-based selection is able to obtain the patches that contains more information (such as those representing hair and eyes). Moreover, during the computation and data storage, the masked patches are not needed, the selected patches are stored as sparse-grid samples, thus the computation and data redundancy can be largely reduced.

3.2 Image Super-Sampling and Reconstruction

Image Super-Sampling algorithms take the LR image as the input and reconstruct the HR image. In this article, for each down-sampling algorithm, one image reconstruction method is proposed accordingly. Therefore, **2 methods** of reconstruction methods are proposed in this project.

3.2.1 Image Super-Sampling based on Deep Learning

In order to reconstruct the HR image from the LR image sampled by Voronoi-Delaunay Triangulation, the deep learning algorithms based on convolutional network are designed and implemented in this project.

The LR image is input into the network and the output is the reconstructed image. In order to minimize the difference between the reconstructed image and the origin image, the Mean-Squared-Error (MSE) loss is used to meter the effect of the reconstruction. The whole structure is depicted in the following figure:

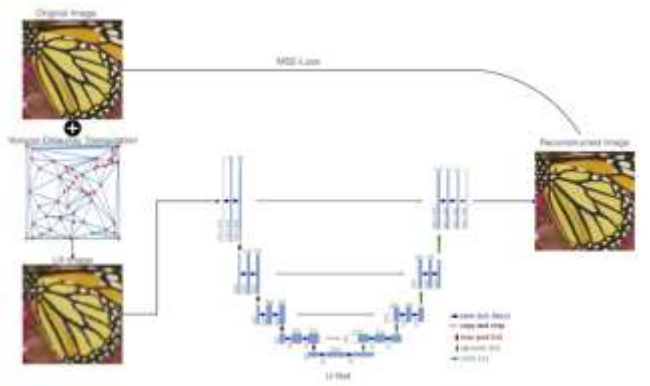

Fig 8: Work-flow of Image super-sampling using DL

In order to find a better network structure for the super-sampling task, the performance of 3 different networks are compared with each other:

1. Super-Resolution Convolutional Neural Network (SRCNN)[30]

SRCNN is the first work that brings deep learning into the field of image super-resolution, which is a very classical super resolution technique.

The structure of the SRCNN Network is shown in the following figure:

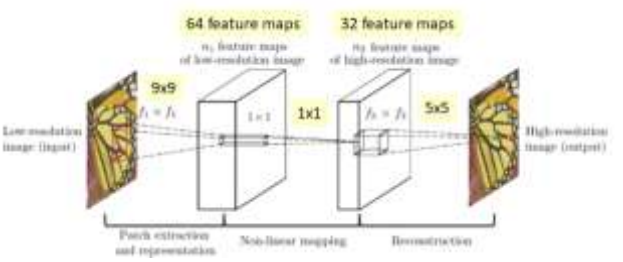

Fig 9: Structure of SRCNN

This is a relatively shallow network, with only 3 parts, patch extraction and representation, non-linear mapping and reconstruction. During the patch extraction and representation operation, the patches of LR image are extracted and represented as high-dimensional vectors. These vectors comprise a set of feature maps, whose number is equal to the dimensionality of the vectors. Then, the non-linear mapping operation maps the high-dimensional vectors into another high-dimensional space, constructing another set of feature maps. Finally, the reconstruction operation aggregates all of the representations to generate the final HR image, which should be as similar to the GT image as possible.

2. Deep Recursive Residual Network (DRRN)[31]

DRRN is a combination of ResNet and SRCNN. In DRRN, the enhanced residual unit structure is recursively learned in a recursive block, and the blocks are stacked to learn the residual between the HR and LR images. The residual image is then added to the input LR image from a global identity branch so as to estimate the HR image.

With Global Residual Learning (GRL) and Multi-path mode Local Residual Learning (LRL) , plus the recursive learning that controls the model size while increasing the depth, up to 52 layers can be achieved in DRRN. The network structure is shown in the following figure.

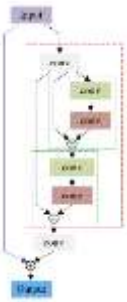

Fig 10: Structure of DRRN

Specifically, the number of residual units and the number of recursive blocks can be manually set, resulting in alterable network depth. The following figure shows the recursive blocks with different residual units.

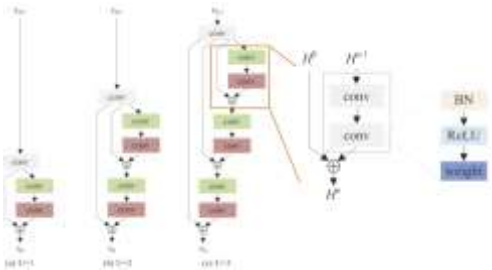

Fig 11: Number of residual units

3. UNet[32]

UNet, which is firstly proposed in 2015 for image segmentation, is build upon is so-called “fully connected convolutional network”. The idea is supplementing the basic contracting network by adding successive layers, in which pooling layers are replaced by up-sampling layers. Therefore, these layers are able to increase the output resolution. Combining the information from the contracting and expansive layers, the network performs better at learning pixel-wise tasks such as segmentation or super-sampling.

During the contraction, the spatial information is reduced while feature information is increased. The expansive pathway combines the feature and spatial information through a sequence of up-convolutions and concatenations with high-resolution features from the contracting path. A large number of up-sampling parts allow it to propagate context information to higher resolution feature layers.

The structure of the network is shown in the following figure. The network consists of a contracting path (left side) and an expansive path (right side), which gives it the U-shaped architecture. The contracting path is a typical convolutional network consisting of convolution layers that extract features, max-pooling layers that down-sample the feature and ReLU activation layers. The expansive path includes up-sampling layers followed by convolution layer (up-convolution), which increases the resolution of the input. The final layer is a simple 1x1 convolutional layer. In all, the whole network includes 23 convolutional layers.

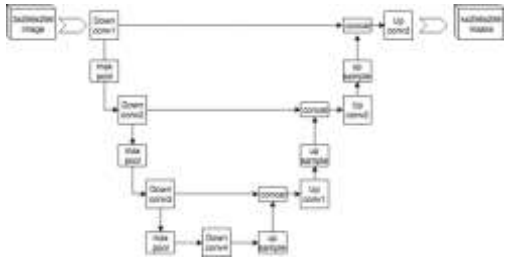

Fig 12 Structure of UNet

3.2.2 Image Reconstruction based on Masked-Auto-Encoder (MAE)

In order to reconstruct the origin image based on the sparse patches, the following problem must be solved: Since the input is no longer the complete image, the super-sampling procedure should not base on models that are only able to extract local features (such as CNNs), then how should we organize the input?

The transformer-based Masked-Auto-Encoder (MAE)[33] is an excellent choice. The input of this model is patches with any amount, and the output is the whole predicted-image. The MAE is a autoencoder that reconstructs the original image given the partial observation of it. During the encoding process, the patches of the image is sent into the network, generating the latent representation of the image and in the decoding process, the representation is decoded back into image, which is the reconstructed image. The architecture of MAE is shown in the following figure:

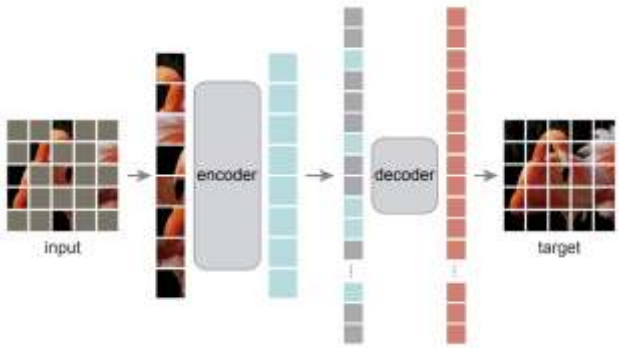

Fig 13: MAE architecture

In detail, the MAE consists of these parts:

1. Masking

   Each image is split into non-overlapping patches, which is similar to Vision Transformer (ViT). Next, using the methods proposed above, a subset of the patches are chosen and others are removed. This approach significantly reduces the data and computation redundancy, and the sparse input makes it possible for designing a large and efficient encoder.

2. Encoding

   The encoding method is similar to that of ViT, the input patches are firstly embedded by a linear projection layer and are added with positional embeddings. Specially, the MAE only uses the chosen patches, meaning that the masked patches are removed from the input. This allows the model to use a relatively large encoder, compared with traditional methods.

3. Decoding

   Different from the encoder, the decoder takes everything into input, including the chosen patches and the masked tokens so as to reconstruct the origin image. In the output, each masked token is a vector learned

by the model, predicting the origin image.

4. Reconstruction Target

The MAE generates a pixel-wise reconstruction for each masked patch. Each value in the output of the decoder represents the pixels of a patch. In order to get the reconstruction image, the output of the decoder is reshaped. In order to evaluate the difference between the reconstruction and the original image, the Mean Squared Error (MSE) loss function is used.

Therefore, the value of each pixel of the origin image can be predicted based on sparse-grid patch samples.

In this article, the target image is reconstructed following the procedure shown in the following figure:

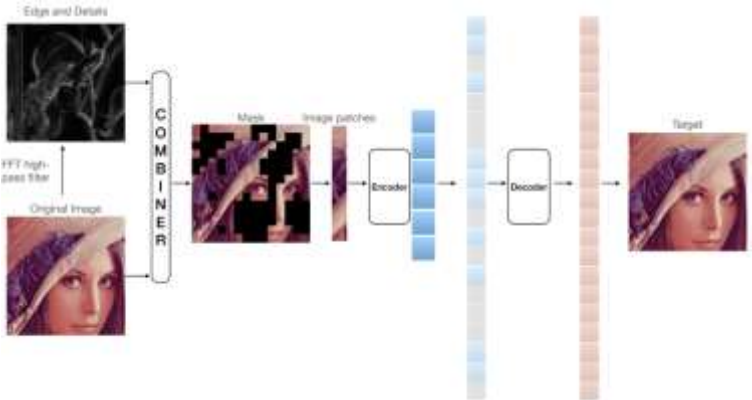

Fig 14: Work-flow of image reconstruction using MAE

Firstly, the original image is input into a FFT high-pass filter, and the edges and details will be retained by the filter.

Secondly, we sparsely sample on the original image by combining the information of the original image and the image details, making sure that the patches with more details are chosen.

Thirdly, the chosen patches are embedded and added with positional embeddings.

Fourthly, the embeddings are sent into the MAE, which extracts the representation of the patches and further predicts the values of the masked patches.

Finally, the predictions are reshaped and the image is reconstructed by combining the chosen patches and the predicted patches.

4 Experiment Results and Analysis

In this section, very detailed experiments regarding to the two methods are performed and analyzed.

4.1 Setup

All of the experiments are performed on a server with 8*TITAN GPUs.

For the image super-sampling algorithms, the dataset is obtained from Set 5, Set 14, BSD 100, Manga 109 and Urban 100, resulting in a train-set with 304 images and a validation-set with 32 images. In order to augment the train-set, the images are randomly flipped both vertically and horizontally, and the RGB channels are randomly permuted. These operations increase the size of train-set from 304 to 7296.

For the image reconstruction based on MAE, the weight of model is loaded from a pre-trained weight trained on ImageNet.

4.2 Image Super-Sampling based on Voronoi-Delaunay Triangulation Samples

In this section, the detailed experiments are performed. To be specific, the performance and results using

SRCNN, DRRN and UNet to super sample the LR images down-sampled by random and FFT down-sampling methods are compared. To thoroughly analyze each method, the down-sampling scale ranges from 3, 10 to 100, resulting in 18 groups of experiments.

The PSNR result of each experiment is shown in the following table.

		UNet	SRCNN	DRRN
X3	FFT	23.56	28.62	**29.31**
	random	24.87	26.67	**28.10**
X10	FFT	22.49	24.68	**25.37**
	random	20.90	22.38	**23.11**
X100	FFT	17.94	17.82	**18.39**
	random	17.77	18.03	**18.19**

Table 1: PSNR results of each experiment

The comparison of the super-resolution results of each network is shown in the following figures.

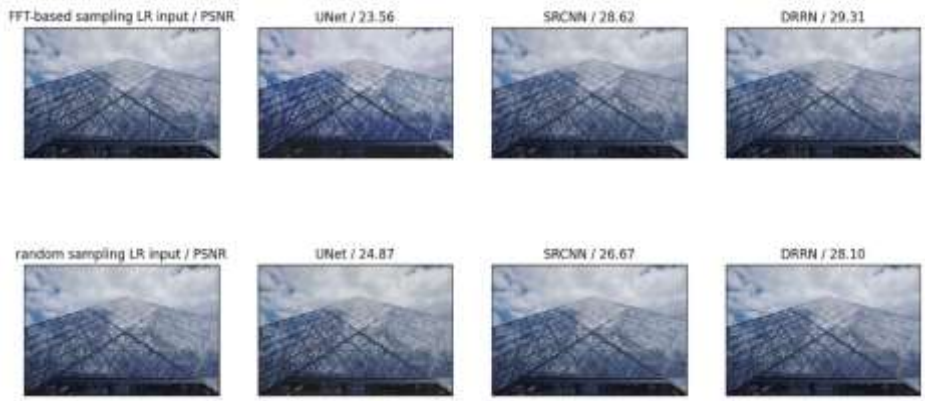

Fig 15: Super-resolution results of each network (3X down-sampling)

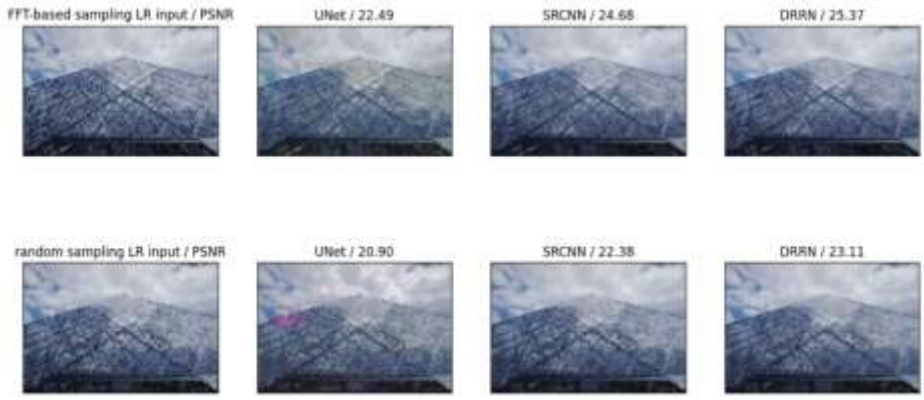

Fig 16: Super-resolution results of each network (10X down-sampling)

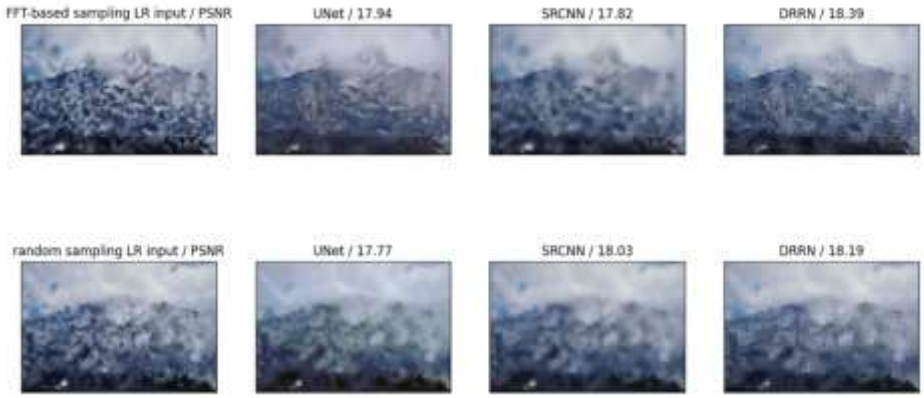

Fig 17: Super-resolution results of each network (100X down-sampling)

The following conclusions can be obtained based on the figures and PSNR results:

1. DRRN outperforms other networks in all situations, meaning that it is the best tested network for this task.
2. Sampling based on FFT enable the LR image to retain more information within the same sample points as random sampling, so it makes it easier for the network to extract the feature from LR images and reconstruct the HR images, resulting in much higher PSNR.
3. Compared with the traditional down-sampling methods such as bilinear interpolation, the down-sampling method proposed in this project pay much more attention to the regions with more details. This result in much more information retained in the LR images, and the detailed features of the origin images are not heavily lost even if the down-sampling rate is as high as 10.

4.3 Image Reconstruction based on Masked-Auto-Encoder (MAE)

In this section, the reconstruction results of the patches are depicted. In order to compare the reconstruction results, the mask-ratio ranging from 0.1 to 0.9 are adopted. The results of the experiments are shown in the following figure:

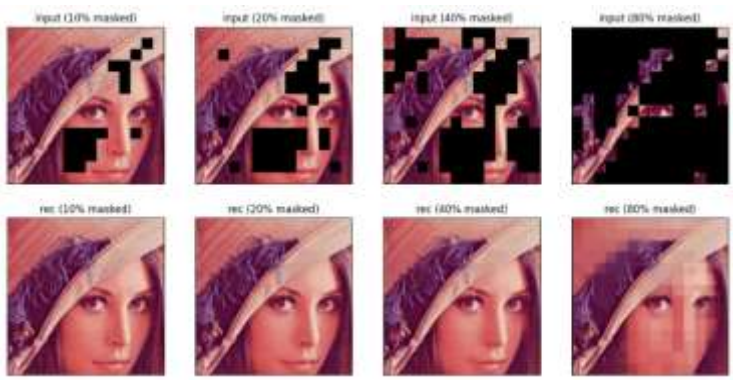

Fig 18: reconstruction effects (FFT-based mask)

Fig 19: reconstruction effects (random mask)

It is shown by the figure that the random masking strategy may cause very bad reconstruction result even if masking ratio is low (20%). On the contrary, the FFT-based masking ratio tend to generate better and more stable results, especially when masking ratio is relatively low.

5 Distinctive or Innovation Points

To sum up, this project includes the following innovations:

1. Down-sampling Methods

   In order to sparsely sample on the origin image, the Voronoi-Delaunay Triangulation and Image Patch Splitting is performed respectively for **sparse-grid** sampling and **sparse-patch** sampling.

   a) Sparse-Grid Sampling

      In the sampling process, using FFT to extract the detailed features of the image, the basic random sampling is improved, resulting in much better LR images within the same down-sampling rate.

   b) Sparse-Patch Sampling

      This sampling method is a novel method for sparse-sampling, which is defined as **Sparse-Patch Sampling** It is similar to that of ViT, which split the image into patches and sparsely sample from them. Besides, FFT is also used in Sparse-Patch Sampling.

2. Image Super-Resolution and Reconstruction Methods

   For sparse-grid sampling and sparse-patch sampling, a method is designed respectively.

   a) Image Super-Resolution

      This method is used for obtaining HR images from LR images down-sampled by sparse-grid sampling. Specifically, SRCNN, UNet and DRRN are tested to obtain the HR images, and the performance of each network on each sampling method are analyzed in detail.

   b) Image Reconstruction

      This method is designed for obtaining the origin image given the patches down-sampled using sparse-patch sampling. To be specific, MAE is used to reconstruct the origin images.

      The origin goal of MAE is representation learning instead of image reconstruction or super-resolution. This project brings this method into the field of image reconstruction from sparse samples, thus being a very distinctive point.

      Moreover, the potential of using MAE in image reconstruction can be further extended. Specifically, it can be utilized in video reconstruction. Since the information of two consecutive frames stay mostly unchanged, during the rendering of the current frame, most information can be directly obtained from

the former frame. The regions with much changed information can be considered as the masks, thus can be predicted by MAE given the other unmasked regions.

6 Supplementary：

Some of the code used in this project are obtained from the following resources, and their hard work are highly appreciated!

1. SRCNN: https://github.com/yjn870/SRCNN-pytorch
2. DRRN: https://github.com/jt827859032/DRRN-pytorch
3. UNet: https://github.com/milesial/Pytorch-UNet
4. MAE: https://github.com/pengzhiliang/MAE-pytorch

References:

[1] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

[2] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, pp. 2180–2188, 2016.

[3] Tejas D Kulkarni, Will Whitney, Pushmeet Kohli, and Joshua B Tenenbaum. Deep convolutional inverse graphics network. *arXiv preprint arXiv:1503.03167*, 2015.

[4] Sjoerd van Steenkiste, Francesco Locatello, Ju ̈rgen Schmidhuber, and Olivier Bachem. Are disen- tangled representations helpful for abstract visual reasoning? *arXiv preprint arXiv:1905.12506*, 2019.

[5] Alessandro Achille, Tom Eccles, Loic Matthey, Christopher P Burgess, Nick Watters, Alexander Lerchner, and Irina Higgins. Life-long disentangled representation learning with cross-domain latent homologies. *arXiv preprint arXiv:1808.06508*, 2018.

[6] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. *arXiv preprint*, 2016.

[7] Hyunjik Kim and Andriy Mnih. Disentangling by factorising. In *International Conference on Machine Learning*, pp. 2649–2658. PMLR, 2018.

[8] Ricky T. Q. Chen, Xuechen Li, Roger Grosse, and David Duvenaud. Isolating sources of disen- tanglement in variational autoencoders. In *Advances in Neural Information Processing Systems*, 2018.

[9] Abhishek Kumar, Prasanna Sattigeri, and Avinash Balakrishnan. Variational inference of disentan- gled latent concepts from unlabeled observations. *arXiv preprint arXiv:1711.00848*, 2017.

[10] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

[11] Zinan Lin, Kiran Thekumparampil, Giulia Fanti, and Sewoong Oh. Infogan-cr and modelcentrality: Self-supervised model training and selection for disentangling gans. In *International Conference on Machine Learning*, pp. 6127–6139. PMLR, 2020.

[12] Insu Jeon, Wonkwang Lee, and Gunhee Kim. Ib-gan: Disentangled representation learning with information bottleneck gan. *arXiv preprint*, 2018.

[13] Wonkwang Lee, Donggyun Kim, Seunghoon Hong, and Honglak Lee. High-fidelity synthesis with disentangled representation. In *European Conference on Computer Vision*, pp. 157–174. Springer, 2020b.

[14] Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Raetsch, Sylvain Gelly, Bernhard Scho ̈lkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. In *international conference on machine learning*, pp. 4114–4124. PMLR, 2019.

[15] Cian Eastwood and Christopher KI Williams. A framework for the quantitative evaluation of disen- tangled representations. In *International Conference on Learning Representations*, 2018.

[16] Abhishek Kumar, Prasanna Sattigeri, and Avinash Balakrishnan. Variational inference of disentan- gled latent concepts from unlabeled observations. *arXiv preprint arXiv:1711.00848*, 2017.

[17] Aapo Hyva¨rinen and Erkki Oja. Independent component analysis: algorithms and applications. *Neural networks*, 13(4-5):411–430, 2000.

[18] Ilyes Khemakhem, Diederik Kingma, Ricardo Monti, and Aapo Hyvarinen. Variational autoen- coders and nonlinear ica: A unifying framework. In *International Conference on Artificial Intel- ligence and Statistics*, pp. 2207–2217. PMLR, 2020a.

[19] David Klindt, Lukas Schott, Yash Sharma, Ivan Ustyuzhaninov, Wieland Brendel, Matthias Bethge, and Dylan Paiton. Towards nonlinear disentanglement in natural data with temporal sparse coding. *arXiv preprint arXiv:2007.10930*, 2020.

[20] Hermanni Ha¨lva¨, Sylvain Le Corff, Luc Lehe´ricy, Jonathan So, Yongjie Zhu, Elisabeth Gassiat, and Aapo Hyvarinen. Disentangling identifiable features from noisy data with structured nonlinear ica. *arXiv preprint arXiv:2106.09620*, 2021.

[21] Aapo Hyvarinen and Hiroshi Morioka. Unsupervised feature extraction by time-contrastive learning and nonlinear ica. *Advances in Neural Information Processing Systems*, 29:3765–3773, 2016.

[22] Ilyes Khemakhem, Diederik Kingma, Ricardo Monti, and Aapo Hyvarinen. Variational autoen- coders and nonlinear ica: A unifying framework. In *International Conference on Artificial Intel- ligence and Statistics*, pp. 2207–2217. PMLR, 2020a.

[23] Aapo Hyvarinen, Hiroaki Sasaki, and Richard Turner. Nonlinear ica using auxiliary variables and generalized contrastive learning. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 859–868. PMLR, 2019.

[24] IlyesKhemakhem,RicardoPioMonti,DiederikPKingma,andAapoHyva¨rinen. Ice-beem: Identifiable conditional energy-based deep models based on nonlinear ica. *arXiv preprint arXiv:2002.11537*, 2020b.

[25] David Klindt, Lukas Schott, Yash Sharma, Ivan Ustyuzhaninov, Wieland Brendel, Matthias Bethge, and Dylan Paiton. Towards nonlinear disentanglement in natural data with temporal sparse coding. *arXiv preprint arXiv:2007.10930*, 2020.

[26] Fialka, Ondirej, and Martin Cadik. "FFT and convolution performance in image filtering on GPU." In Tenth International Conference on Information Visualisation (IV'06), pp. 609-614. IEEE, 2006

[27] Tian, Chunwei, Qi Zhang, Guanglu Sun, Zhichao Song, and Siyan Li. "FFT consolidated sparse and collaborative representation for image classification." Arabian Journal for Science and Engineering 43, no. 2 (2018): 741-758.

[28] Andrew Edelsten, Paula Jukarainen, and Anjul Patney. 2019. Truly next-gen: Adding deep learning to games and graphics. In NVIDIA Sponsored Sessions (Game Developers Conference)

[29] Lei Xiao, Salah Nouri, Matt Chapman, Alexander Fix, Douglas Lanman, and Anton Kaplanyan. 2020. Neural Supersampling for Real-time Rendering. ACMTrans. Graph. 39, 4, Article 142 (July 2020), 12 pages. https://doi.org/10. 1145/3386569.3392376

[30] Ma, Xiaofeng, Youtang Hong, Yongze Song, and Yujia Chen. "A super-resolution convolutional-neural-network-based approach for subpixel mapping of hyperspectral images." IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 12, no. 12 (2019): 4930-4939.

[31] Tai, Ying, Jian Yang, and Xiaoming Liu. "Image super-resolution via deep recursive residual network." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3147-3155. 2017.

[32] Ronneberger, Olaf, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation." In International Conference on Medical image computing and computer-assisted intervention, pp. 234-241. Springer, Cham, 2015.

[33] He, Kaiming, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. "Masked autoencoders are scalable vision learners." arXiv preprint arXiv:2111.06377 (2021).

时间安排与分工统计表

<table>
<tr><td colspan="4">组员信息（含组长）</td></tr>
<tr><td>学生姓名</td><td>窦铱明</td><td>学　　号</td><td>519021910366</td></tr>
<tr><td>项目分工</td><td colspan="3">两种降采样方法的设计与实现；
针对两种降采样方法的 4 种超分辨率与图像重建算法的设计与实现；
撰写报告 Methods、Experiment 部分；
答辩 PPT 修改</td></tr>
<tr><td>学生姓名</td><td>杨晴</td><td>学　　号</td><td>519021910756</td></tr>
<tr><td>项目分工</td><td colspan="3">相关文献查找；
撰写报告 Related Works 部分；
答辩 PPT 制作</td></tr>
<tr><td>时间安排/
Schedule</td><td colspan="3">选题：2021.10.30
方案制定：2021.10.30
试验研究：2021.10.30~2021.11.28
数据处理：2021.11.28~2021.12.1
研制开发：2021.12.1~2021.12.23
撰写总结报告：2021.12.23~2021.12.31</td></tr>
</table>