Defocusing and Deblurring by Using with Fourier Transfer

AKIRA YANAGAWA and TATSUYA KATO

1. Introduction

Image data may be obtained through an image system, such as a video camera or a digital still camera. When we obtain an image, some noise can be added to it. This noise depends on an image system and often interferes with a sampling trait from this image. In the case of defocusing, this is caused by the rays, which are focused on a single point in an ideal system, and are slightly spread out [Hor86]. In this paper, we will try to make a defocused gray-scale image artificially. Making use of this theorem, we will try to debulur a gray-scale picture without its original image. We will also try to expand the experiment to a color image.

2. The Definition of Defocus and Deblur

Defocusing can be modeled as Point Spread Function (PSF) [Ros76]. By using PSF, defocusing is defined as the convolution of the data of an image and PSF (Fig. 1 and Fig. 2.1).

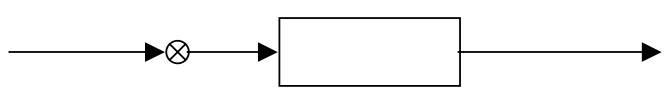

Fig. 1

$$g(x,y)=\int_{-\infty}^{\infty} h(x-\xi, y-\eta) f(\xi,\eta) d\xi d\eta \qquad (2.1)$$

Where h(x, y) is PSF, f(x, y) is an ideal image, and g(x, y) is a blurred image.[Sai93]

In general, it is based on Gaussian model although there is much PSF. Eqn. (2.2) denotes 2-D Gaussian, and Blur PSF (Gaussian) and Ideal PSF are shown in Fig. 2.[Hor86]

$$h(x,y)=\frac{1}{2\pi\sigma^{2}} e^{-\frac{(x^{2}+y^{2})}{2\sigma^{2}}} \qquad (2.2)$$

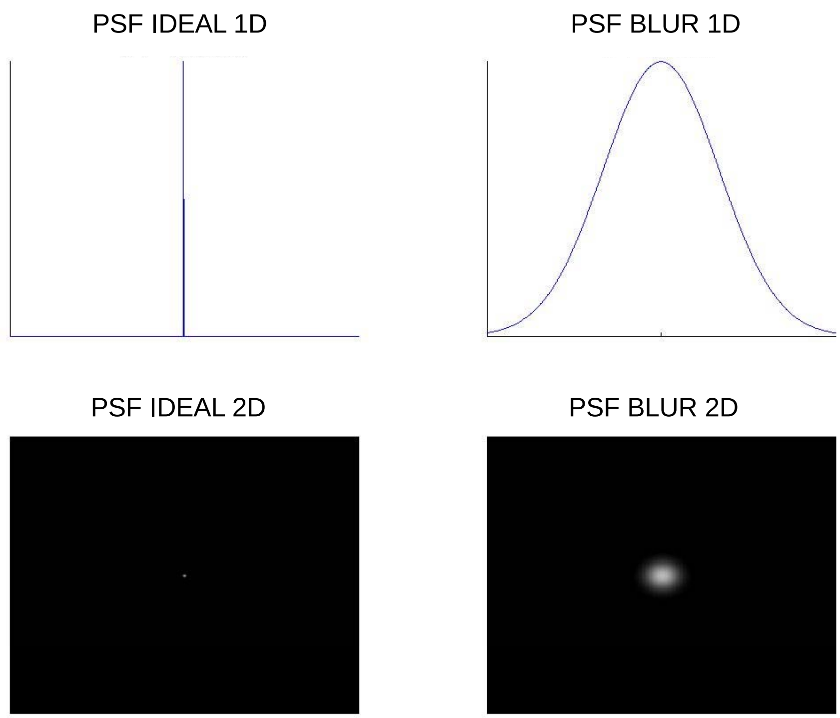

Fig. 2 PSF BLUR $\mu=0$ $\sigma=10$

In ideal PSF, the rays are not spread. In 1-D, it is δ-function, and in 2-D, it is a point. On the other hand, in Gaussian 2-D PSF, the rays spread from the center of the lens to outside. This causes blurring.

To decrease the amount of the calculation, we have used the Fourier transform. Eqn. (2.3) denotes the Fourier transform of Eqn. (2.2). This shows that the Fourier transform of Gaussian is also Gaussian. [Hor86] We can transform Eqn. (2.2) to Eqn. (2.4) by using Eqn. (2.3). [Sai93]

$$H(u,v) = e^{-\frac{1}{2}(u^2+v^2)\sigma^2} \qquad (2.3)$$

$$G(u,v) = H(u,v)F(u,v) \qquad (2.4)$$

Where G(u, v) is the Fourier transform of a blurred image, H(u, v) is the Fourier transform of a blurred image, and F(u, v) is Fourier transform of an ideal image. H (u, v) is a low-pass filter (see Fig. 3 showing the magnitude of H (u, v)).

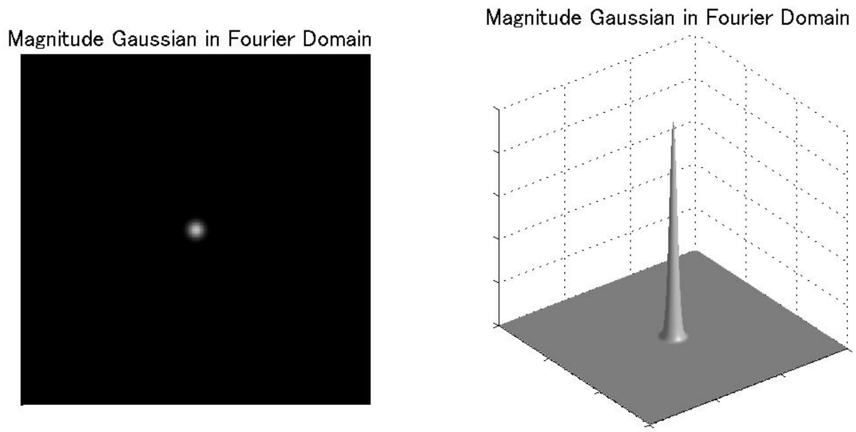

Fig. 3

We can also deblur an image by using the inverse function of Eqn. (2.4). [Ros76]

$$F(u,v) = G(u,v)/H(u,v) \tag{2.5}$$

M (u, v) denotes the inverse function of Eqn. (2.6).

$$M(u,v) = 1/H(u,v) \tag{2.6}$$

However, if H (u, v) is 0 or close to 0, F (u, v) will be infinity or have a large value. To solve this problem, we applied the Wiener filter, denoted by Eqn. (2.7).

$$M(u,v) = \frac{H(u,v)}{|H(u,v)|^2 + \Gamma} \tag{2.7}$$

Since to prove Eqn. (2.7) is a long and complex process, I have omitted the proof (it shows in [Ros76]). However, we found out intuitively that M (u, v) will not diverge at any H(u, v) because of Γ. This Γ denotes the S/N ratio of G(u, v), and at present, the method of estimating Γ without the information of the image. Therefore, we have to decide Γ reciprocally. The Wiener filter, however, is effective to deblur an image by comparing it with Eqn. (2.6) [Har68].

We have attached Fig. 4 which shows the magnitude of both Weiner filter and an inverse function. From Fig. 4, the Wiener filter is also more stable than an inverse function.

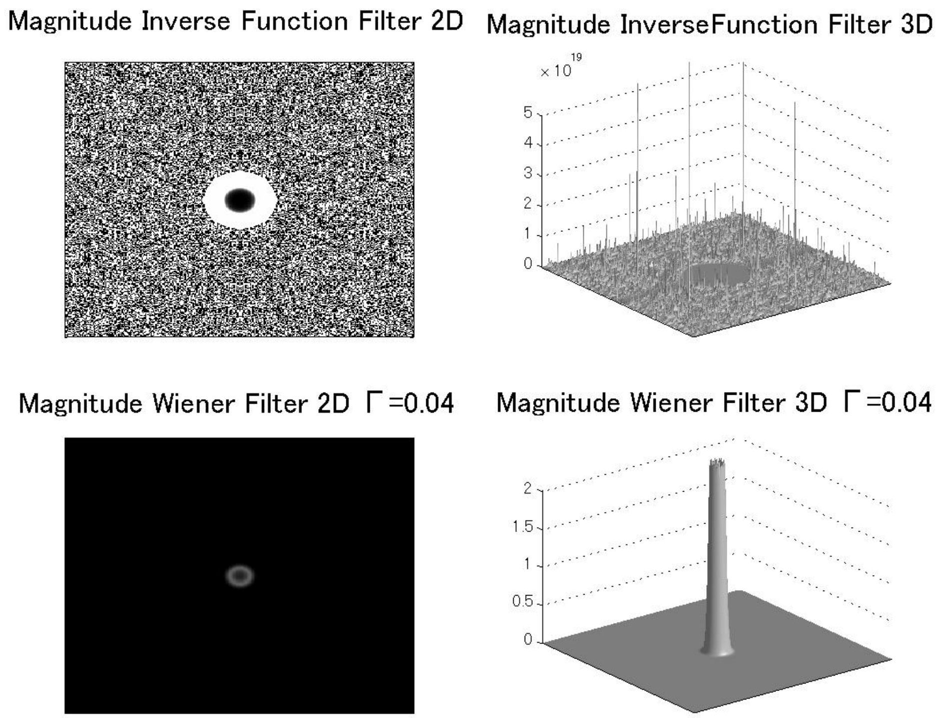

Fig. 4 Magnitude of Inverse Function Filter and Weiner Filter (σ =10, Γ=0.04)

3. Composition and separation of a color

The method stated in Section 2 is only for gray-scale images. To make it practical, we have considered about the way to apply it to color images.

In general, the color picture consists of Red, Green, and Blue (RGB). However, it is possible that the processing causes color shift since each plane has the information of color. Thus, to avoid color shift, we have extracted a gray-scale image by using HSV color space. HSV space has three parameters: hue, saturation, and value. We have transformed RGB into HSV by using rgb2hsv function of MATLAB. After that, we processed V, which has a gray-scale value, and mixed it with H and S (Fig. 5).

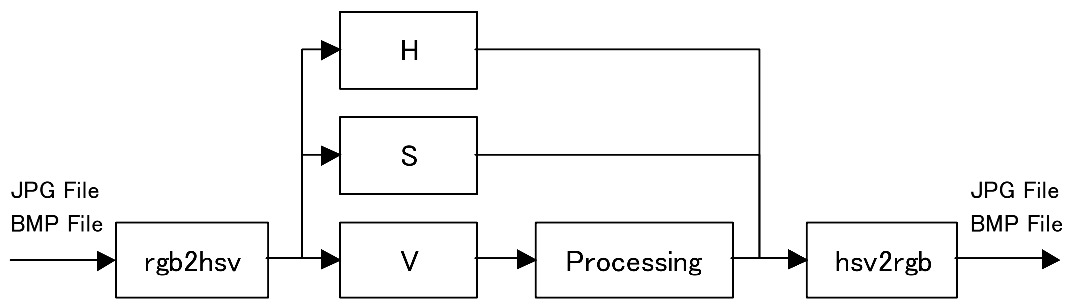

Fig. 5

In this report, we use HSV. Yet, if the application is focused on gray-scale images, YIQ or YUV should be selected because V in HSV[1] is (R+G +B)/3. It has been said that people do not feel that red and blue have the same degree of brightness. This shows formulas to change RGB to YIQ or YUV in (3.1). Y denotes brightness.

$$
\begin{aligned}
Y &= 0.299R + 0.587G + 0.114B \\
I &= 0.596R - 0.274G - 0.322B \\
Q &= 0.212R - 0.523G + 0.311B \\
\\
Y &= 0.299R + 0.587G + 0.114B \\
U &= -0.147R - 0.289G + 0.437B \\
V &= 0.615R - 0.515G - 0.100B
\end{aligned}
\tag{3.1}
$$

[1] There are a lot of formulas to change RGB into HSV. We did to denote the formula because we don't know which formula is used byMATLAB.

4. Experiments and Results

4.1 Procedure of the Experiment

We experimented according to the following procedure.

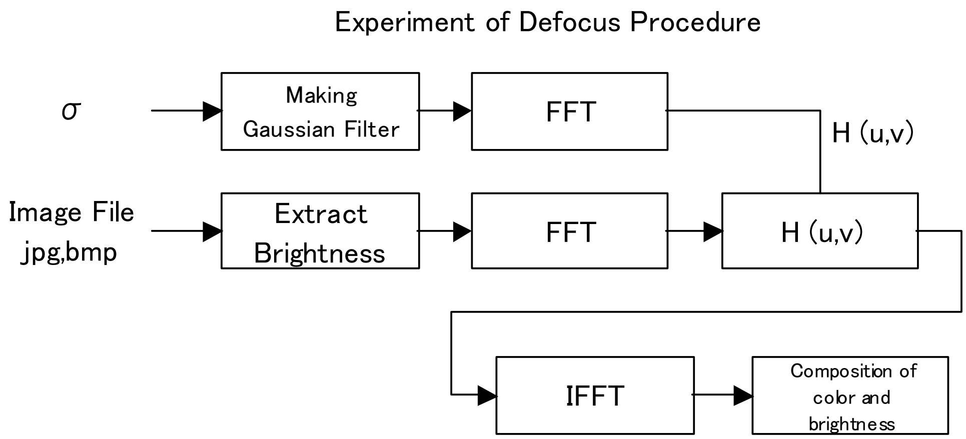

Fig. 6

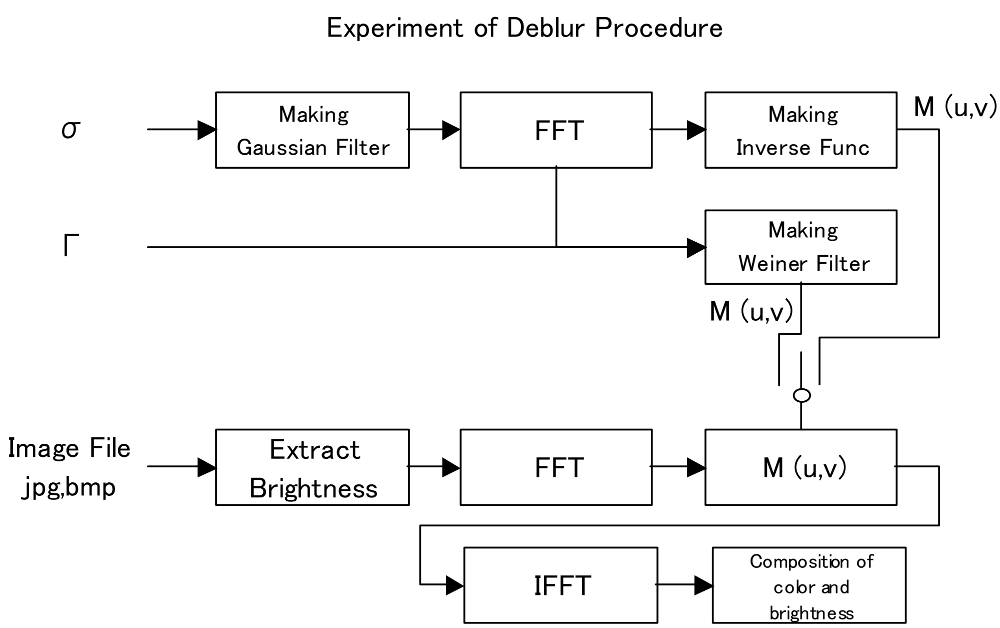

Fig. 7

4.2 Program for experiments

At this time, we use “GUIDE” which is GUI function in MATLAB in order to change the parameters. The program we used for this experiment is shown in Fig. 8.

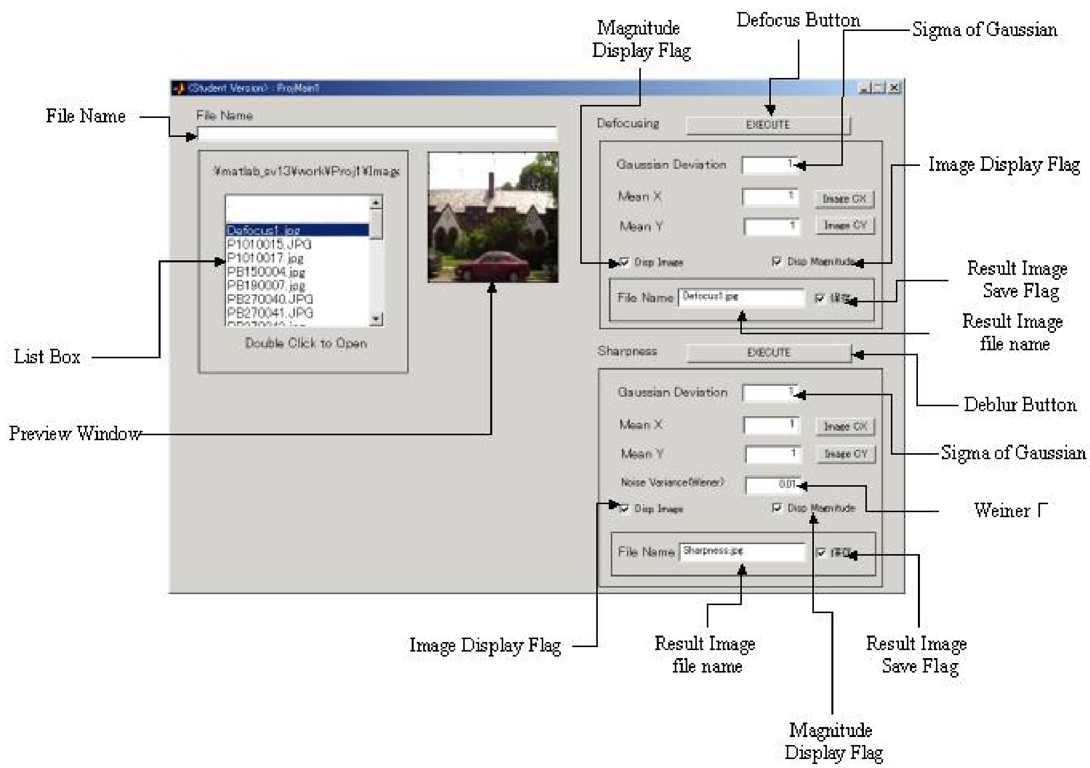

Fig. 8

Defocusing

1. Select image from List Box (DOUBLE CLICK)
2. Input parameter, and select check box.

   Disp Image Check Box

      The result image is displayed on new window.

   Disp Mag Check Box

      The magnitude of the defocus filter is displayed on the new window.

   Save Check Box

      The result image is saved in the filed named in the edit box. File type is JPG or BMP. It selects file type automatically by reading the file name extension.

3. Click Defocus Execute button.

Deblurring

1. Select image from List Box (DOUBLE CLICK)
2. Input parameter, and select check box.

    Disp Image Check Box

        The result image is displayed on the new window.

    Disp Mag Check Box

        The magnitude of the defocus filter is displayed on the new window.

    Save Check Box

        The result image is saved in the filed named in the edit box. File type is JPG or BMP. It selects file type automatically by reading the file name extension.

3. Click Sharpness Execute button.

4.3 Defocus

We confirmed that the approximation of defocus could be obtained with a Gaussian filter.

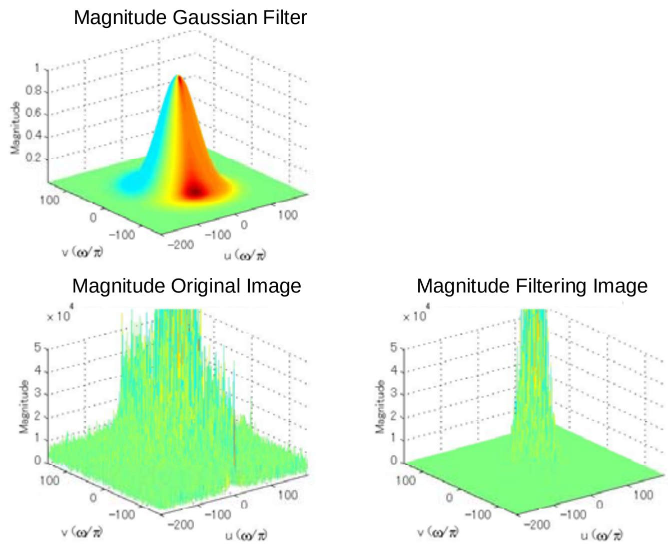

Fig. 9 Magnitude Fourier Domain (σ =1.5)

Fig. 10 Original Image vs. Filtering Image (σ =1.5)

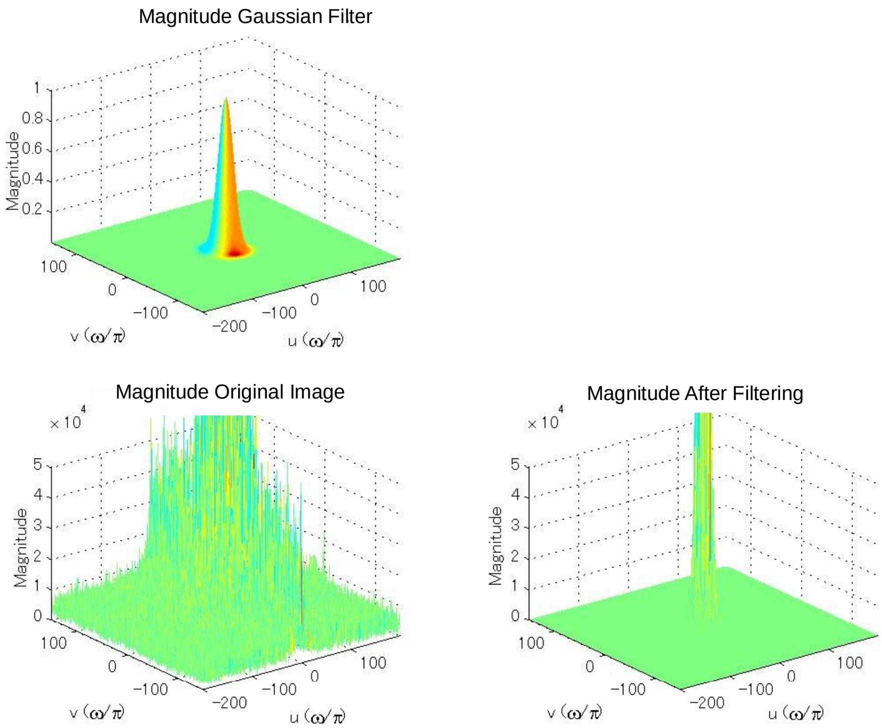

Fig. 11 Magnitude Fourier Domain (σ =5.0)

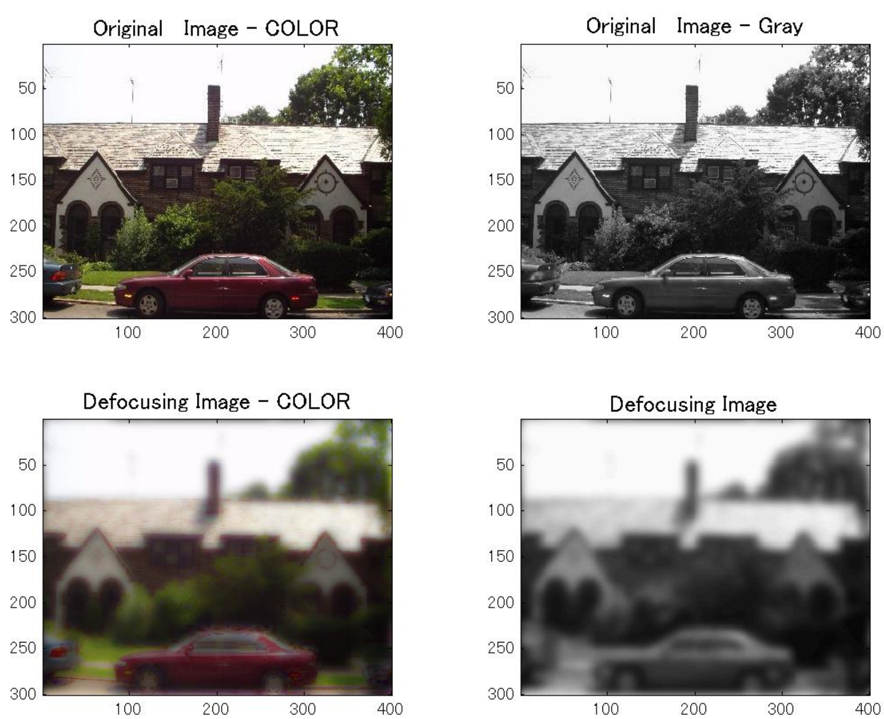

Fig. 12 Original Image vs. Filtering Image (σ =5.0)

At this time, the image shown in Fig. 10 and Fig. 12 was passed through the Gaussian filter in the Fourier domain with $\sigma=1.5$ and $\sigma=5$ respectively. In the 2-D Fourier domain, at the center (u=0, v=0), the frequency is 0. In proportion to the distance from the origin (u=0, v=0), the frequency becomes high. Regarding these facts, the Gaussian filter works as a low-pass filter in Fig. 9 and Fig. 11. As a result, the image was defocused. More frequency will be cut when σ increases. Because of that, the image defocused with $\sigma=5.0$ is more blurred than the image defocused with $\sigma=1.5$.

4.4 Deblur

We tried to deblur images that we have no information of original (focused) images for. Because of that, we decided the suitable parameters of each image by trial and select.

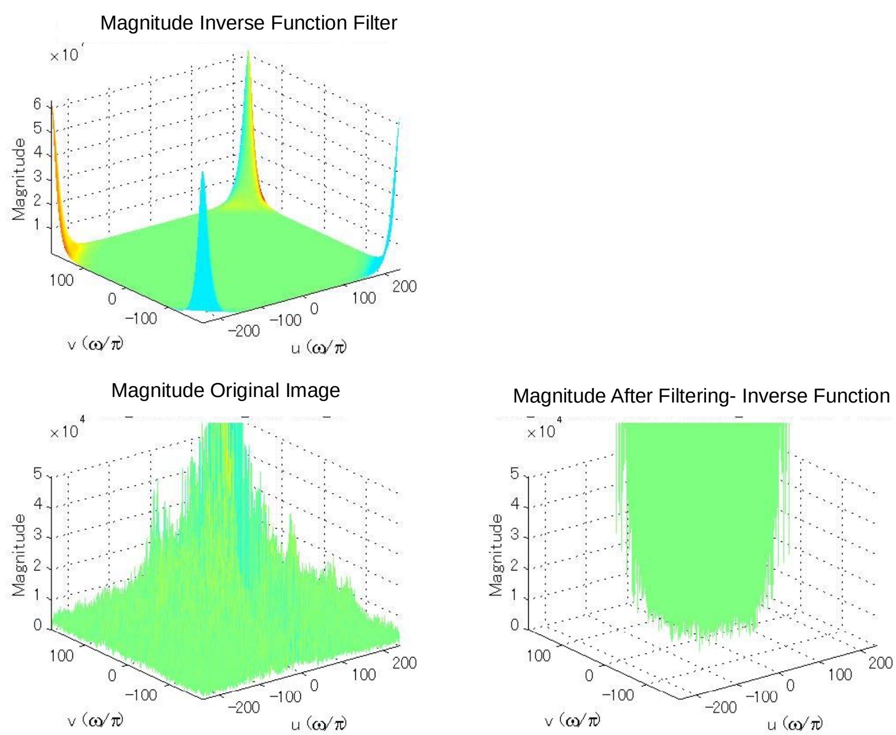

Fig. 13 Magnitude Inverse Function Filter Fourier Domain ($\sigma=1.4$)

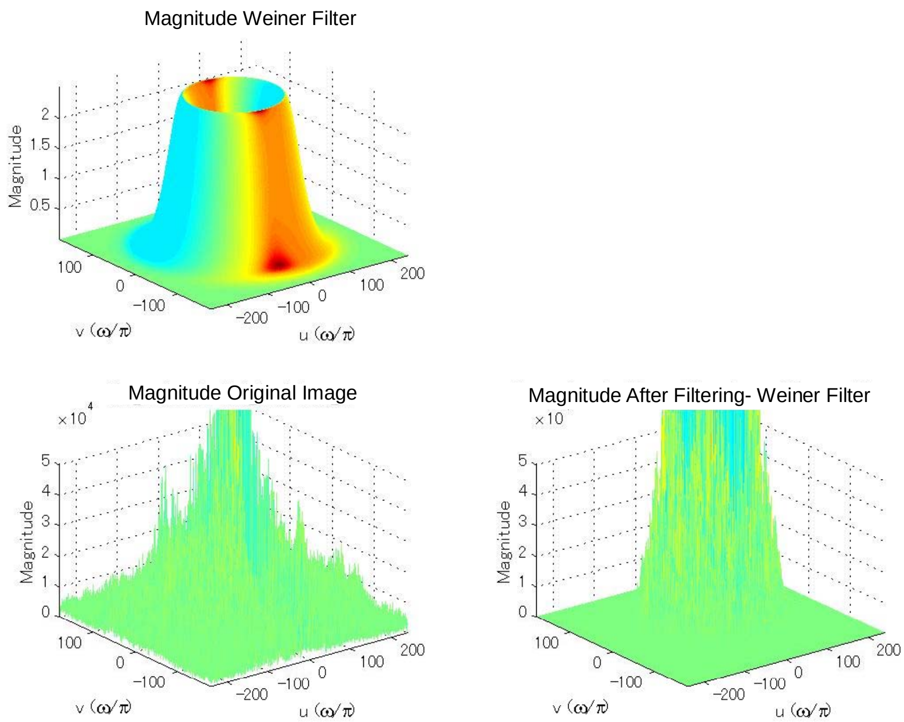

Fig. 14 Magnitude Weiner Filter Fourier Domain (σ =1.4, Γ=0.04)

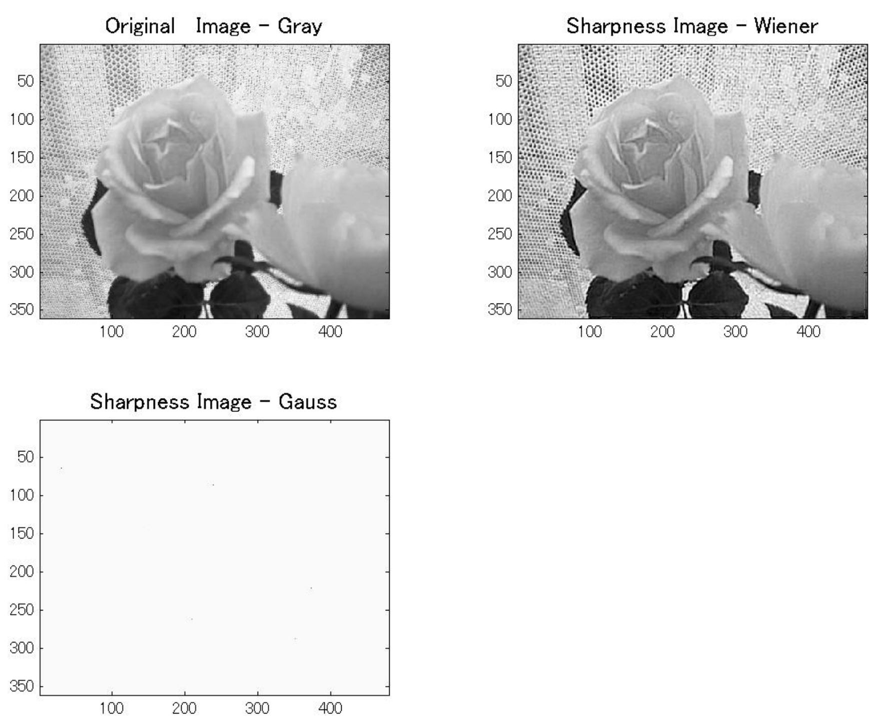

Fig. 15 Original Image vs. Filtering Image (Gray Scale) (σ =1.4, Γ=0.04)

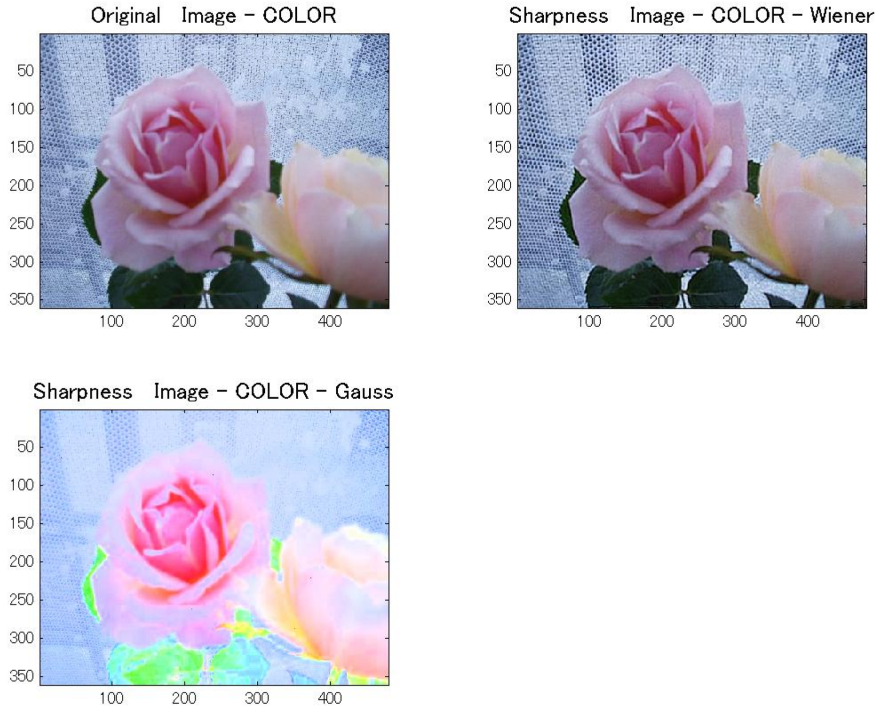

Fig. 16 Original Image vs. Filtering Image (color) (σ =1.4, Γ=0.04)

From the result of the magnitudes, the Inverse Function Filter has diverged at the high frequency area in the Fourier domain space. Because of this, the image passed through the filter was saturated. In contrast, because of Γ, the magnitude of the Wiener filter was restricted so that it may not be 0 or around 0. Also, from Fig. 15 and Fig. 16, after the Weiner Filter passed through, the image became sharper than before.

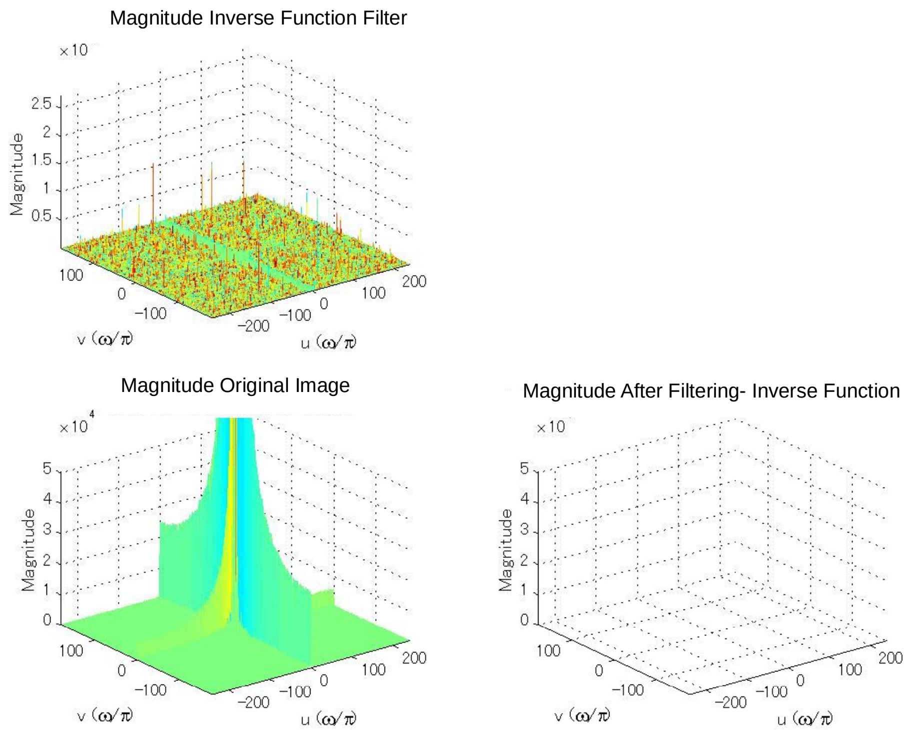

Fig. 17 Magnitude Inverse Function Filter Fourier Domain (σ =24.0)

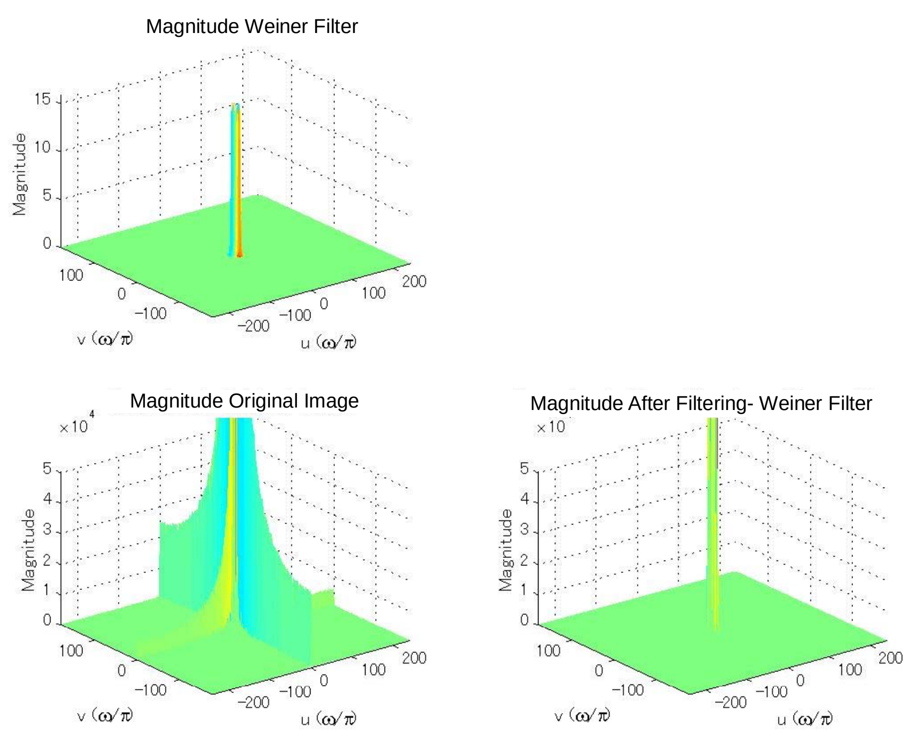

Fig. 18 Magnitude Weiner Filter Fourier Domain (σ =24.0, Γ=0.001)

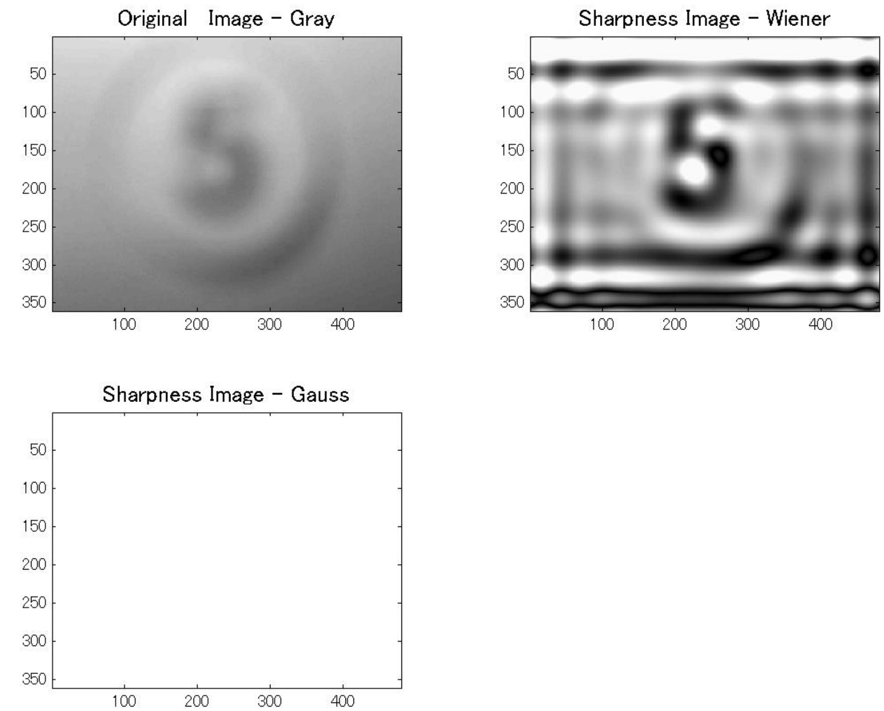

Fig. 19 Original Image vs. Filtering Image (Gray Scale) (σ =24, Γ=0.001)

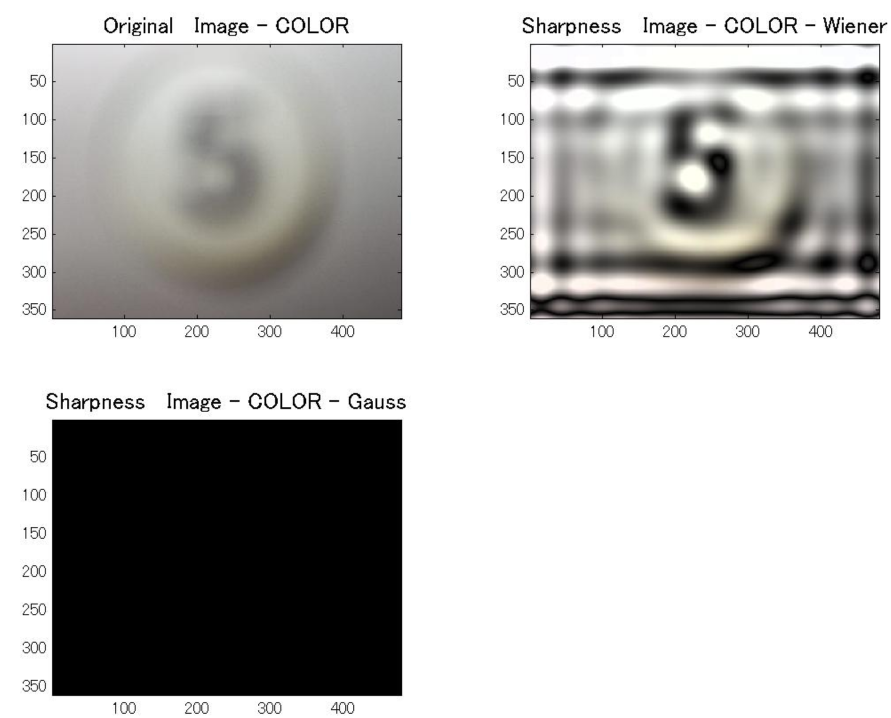

Fig. 20 Original Image vs. Filtering Image (color) (σ =24, Γ=0.001)

In this case, we tried to construct numerical information from the blurred image. If σ is increased, the picture will lose its original texture. However, Fig.19 and Fig.20 show that required information was acquired by emphasizing images' edges (high frequency area) by this method. It is useful to process images, such as the Optical Character Reader (OCR).

5. Conclusion

Blurred images can be made by using Gaussian filter, and the degree of the blur depends on its standard deviation (or variance). Even if we don't know the original (focused) image, we can get rid of the blur from the image by using the Weiner filter. The Wiener filters can also enhance the edge image. This is effective in processing images like the OCR.

6. Further Work

We tried one method of deblurring. However, we have to repeatedly search for suitable parameters (σ and Γ); moreover, noises of images are not the only blurs but also additive noise and other noise. If the original image is known, some methods of construct image, by using posterior probability, are suggested [Sai93] [Ros76]; yet, the method of deblurring from an unknown original image and noise distribution is still not established. In the future, we will try to establish the decision method of σ and Γ and apply this to all unknown images.

Hor86	Horn, B. *ROBOT VISION*, McGraw-Hill, pp. 126-128, 1986
Ros76	Rosenfeld, A. and Hak, A. *Digital Picture Processing*, pp. 209-228
Sai93	Saitoh, T. *Image Processing Algorithm*, pp. 68-84
Har68	Harris, J Potential and limitations of techniques for processing linear motion –degraded imagery, NASA Publ. SP-192, pp. 131-138, 1968