# Development of Manga Panel Cleaner for Digitizing Classic Manga using MATLAB

Reginald Geoffrey L. Bayeta IV
*De La Salle University*
Manila, Philippines
reginald_geoffrey_bayetaiv@dlsu.edu.ph

Kyle Jomar C. Megino
*De La Salle University*
Manila, Philippines
kyle_megino@dlsu.edu.ph

Angelo Jose Teodorico D. Parco
*De La Salle University*
Manila, Philippines
angelo_jose_parco@dlsu.edu.ph

***Abstract*— This paper presents the design and development of Mange Panel Cleaner, an application for automatically cleaning manga panels. The application accepts a raw scan to process and a manually cleaned image for measuring performance metrics. Using the raw scan, the program then cleans the panel. This processed image alongside the raw scan and the manually cleaned image are then used to calculate performance metrics. Image sharpening, de-noising, filtering, and intensity leveling are used in the process of automatic manga panel cleaning. Additionally, performance metrics such as elapsed time, peak signal-to-noise ratio, mean square error, image enhancement factor, and structural similarity index are used as quantitative metrics for performance. Moreover, the measured performance metrics are used to measure the quality of the image and to predict the perceived quality of the image. Time elapsed will be used to measure the speed of the system as a method of automatically cleaning manga panels. To reinforce the quantitative metrics measured, a survey was held to obtain data about the perceived quality. Survey results from a sample of thirty-two show that the manually cleaned images and the automatically cleaned panels is similar or comparable to one another, where sample slightly favors the automatically cleaned panels. For future developments, this study can be extended/complemented by the implementation of batch processing of images to improve user experience alongside more sophisticated image processing techniques to improve the resulting image quality.**

***Keywords—manga cleaning, image processing, denoising, digitization, automation***

## I. INTRODUCTION

Manga are comics that originated from Japan. It is one of the most popular entertainment media in the country with a wide range of audiences from children to adults. Manga also contributes 30% to 45% of Japan's annual publications. Originating from the 1900s, it began gaining traction during the 1960s with the emergence of artists such as Tezuka Osamu, Ishinomori Shotaro, and Chiba Tetsuya and the release of classics such as *Astro Boy* and *Cyborg 009* [1]. Manga publishers such as *Weekly Shounen Jump*, *Kadokawa Shouten*, and *MediaWorks* published popular works such as *Naruto*, *Neon Genesis Evangelion*, *Cowboy Bebop*, and *One Piece*.

With the improvements in technology over the recent years, manga transitioned to being drawn digitally from being hand-drawn. Classic manga, on the other hand, were manually drawn and do not have any digital master copies. Classic manga are less popular due to their scarcity. Old and manga are only available in print and paperback, which makes them inaccessible to a larger audience. Digitizing classic manga allows them to be stored, archived, and be shared, and accessed through the internet.

When classic manga are digitized, they are usually scanned from old copies, which may have a degraded quality, damage, and unwanted artifacts that may persist after scanning. To resolve this, editors usually import scanned images and manually cleaned them using software, which would take some time, especially for complex images and panels. Another option is to use image processing which is faster and more consistent.

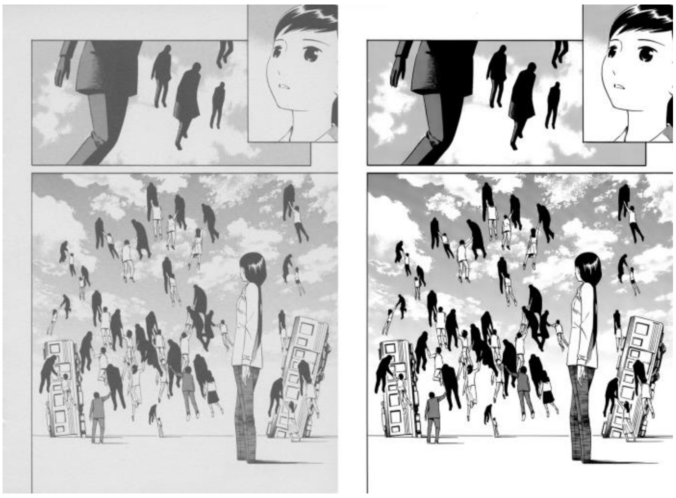


Fig. 1. A sample of a manga panel before and after cleaning [2]

Several studies aim to improve image quality and reduce image noise in the past using image processing techniques. These include Support Vector Regression, Non-Local Means algorithm, Graph-Based Frequency Domain Low-pass Filtering, Dynamic Thresholding, and Wavelet Transform Lifting [3]–[7]. Other image processing techniques include Grid Warping, Edge-error Based Image Sharpening, Contrast Enhancement, High Boost Filtering, and a combination of Sharpening, Filtering, and Contrast Enhancement. [8]–[12].

In this paper, we present Manga Panel Cleaner (MPC), an application developed in MATLAB to automatically clean manga panels for digitization. The application is developed using built-in functions provided within MATLAB as well as utilizing the denoising techniques discussed by [13].

Cleaning manga panels require the use of photo editing software and take an amount of time especially in panels that contain complicated images and shapes. The MPC allows for fast panel cleaning with comparable quality as manually cleaned panels through the use of image processing functions to improve the appearance of scanned manga panels to make them more readable and appealing for readers.

## II. METHODOLOGY

### A. Application Design

In designing MPC as an automatic mange panel cleaning system, the following has been considered in its planning stages, namely:

1. Ease of use. The MPC can be utilized by users without extensive knowledge in image editing and manga panel cleaning.
2. Straightforward interfaces. The functions of the MPC are streamlined and self-explanatory. Additionally, the user does not need extensive knowledge of the application to use it to its fullest.
3. Consistent and bug-free usage. The program only functions when the necessary inputs are available. Additionally, it has a reset function to revert the program to its initial state when necessary or desired.

Manga Panel Cleaner, as an automatic mange panel cleaning system, aims to streamline the process of cleaning manga panels. What would normally be done manually can be done automatically.

## B. *Algorithm and Block Diagram*

Algorithm 1: Raw image to cleaned image
Input: uint8[][] raw_image
Output: uint8[][] cleaned_image, double te, peaksnr, mse, ief, ssim

1. Load “net.mat”
2. Let te = 0
3. *tic*
4. cleaned_image = Sharpen raw_image
5. te = te + *toc*
6. *tic*
7. Adjust leveling of cleaned_image
8. te = te + *toc*
9. *tic*
10. Denoise cleaned_image using DenoiseCNN
11. te = te + *toc*
12. *tic*
13. Put cleaned_image through median filter
14. te = te + *toc*
15. *tic*
16. Re-adjust leveling of cleaned_image
17. te = te + *toc*
18. *tic*
19. Re-sharpen cleaned_image
20. te = te + *toc*
21. *tic*
22. Apply Gaussian Filter to cleaned_image
23. te = te + *toc*
24. Get the psnr of cleaned_image and manually cleaned_image
25. Get mse of cleaned_image and manually cleaned_image
26. Get the ief of cleaned_image and manually cleaned_image
27. Get the ssim of cleaned_image and manually cleaned_image

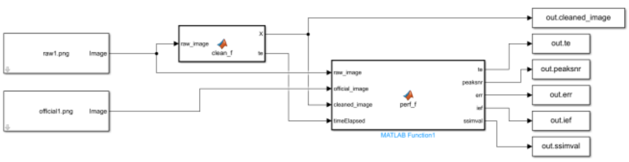



Fig. 2. Block diagram of Manga Panel Cleaner

The block diagram described in Figure 1 shows the inputs and outputs of the system. The system requires two inputs: (1) The raw image for image processing and measurement of performance metrics, and (2) the manually cleaned image for measurement of performance metrics. The image processing block performs image sharpening, filtering, and de-noising on the raw image to produce a cleaned image. The time to complete all these processes is measured. The output of the image processing block is the cleaned image and the total time elapsed for cleaning the image. The performance measurement block, then accepts the raw image, the manually cleaned image, the cleaned image, and the measured time elapsed to obtain the desired performance metrics. The system then outputs the cleaned image and the performance metrics.

## C. *Measuring System Performance*

In this section, metrics such as *time elapsed, peak signal-to-noise ratio, mean-square error, image enhancement factor, structural similarity index, and perceived quality* will be used to measure the performance of the system. Four raw scans will be used and put through the system to measure the aforementioned performance metrics.


Fig. 3. The four raw manga scans that will be used as system inputs [14]

1. Time Elapsed
   This refers to the total time taken to process the image. This metric will show how quickly the system can process images. To capture the time elapsed, the *tic* and *toc* functions will be used. *tic* begins the timer while *toc* returns the amount of time elapsed since *tic* was called. This will be called every time an image processing function is called, with the returned value of *toc*. This will be called every time an image processing function is called.

$$T_{elapsed} = T_{elapsed} + toc \tag{1}$$

   The returned value of *toc* will then be added to the currently measured elapsed time, as described by (1).

2. Peak Signal-to-noise Ratio
   This refers to the ratio between the peak amplitude of the power of the image signal and the noise or mean-square error of the processed file. This metric quantifies the quality of the image processing. Greater values are desired. Equation (2) shows the retrieval of peak signal-to-noise ratio in the decibel (dB) notation.

$$PSNR = 20 * log_{10}\left(\frac{MAX_i}{\sqrt{MSE}}\right) \tag{2}$$

3. Mean Square Error
   This refers to the average variance or squared standard deviation between the processed or 'cleaned' image and the manually cleaned image. This is a metric to quantify the difference or deviation of the processed image from the manually cleaned image. Lesser values are desired.

$$MSE = \frac{1}{n} * \sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2 \tag{3}$$

4. Image Enhancement Factor
   Image enhancement factor is a metric for quantifying how much the processed image was enhanced as compared to the original, unprocessed image, with respect to the manually cleaned image. Equation (4) shows that this is taken by dividing the *MSE* of the processed image and the manually cleaned image with the *MSE* of the raw image and the manually cleaned image.

$$IEF = \frac{MSE_{processed}}{MSE_{raw}} \tag{4}$$

5. Structural Similarity Index
   This refers to the attempted quantification of the similarity between two imaged. It attempts to predict the perceived quality of the processed image. Equation (5) gets the structural similarity of two images, *x* and *y*.

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x{}^2 + \mu_y{}^2 + c_1)(\sigma_x{}^2 + \sigma_y{}^2 + c_2)} \tag{5}$$

   where:
   $\mu_x$ is the average of $x$,
   $\mu_y$ is the average of $y$,
   $\sigma_x$ is the covariance of $x$,
   $\sigma_y$ is the covariance of $y$,
   $\sigma_{xy}$ is the covariance of $x$ and $y$,
   $c_1$ and $c_2$ are stabilizing variables in case of weak denominators.

6. Perceived Quality
   Using a survey, the perceived quality of the processed image will be measured. The survey will query which between the manually cleaned files and processed image is the better-looking option.

## III. SIMULATION

### *A. Manga Panel Cleaner Program*

The program has three panels visualizing the raw scan, manual clean, and resulting image. Above the former two are buttons that allow the user to import each of the raw scans and manually cleaned files.

Below the panels are buttons labeled 'START', 'SAVE', and 'CLEAR'. 'START' will begin the image processing. It does nothing if either the raw or manually cleaned files are not imported. When the image is processed, a progress bar shows the stage of processing that the raw scan is going through. When the image processing is done, the bottom-most table shows the performance metrics of the cleaned image, and the 'SAVE' button lights up, allowing the user to save the cleaned panel. The 'CLEAR' button removes all visualization and performance metric data, reverting the program to its initial state.

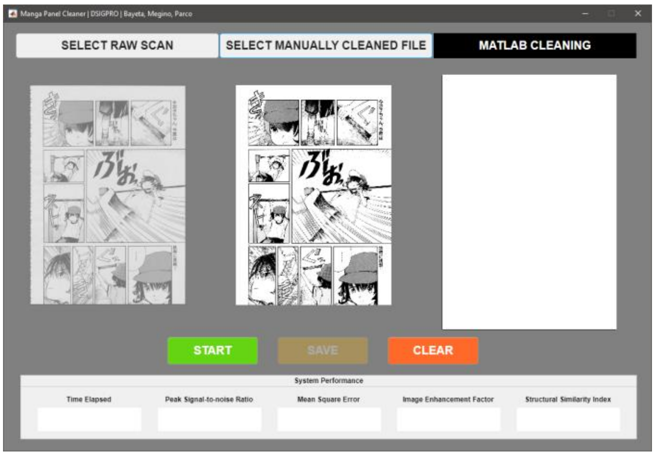



Fig. 4. MPC with the selected raw scan and manually cleaned files.

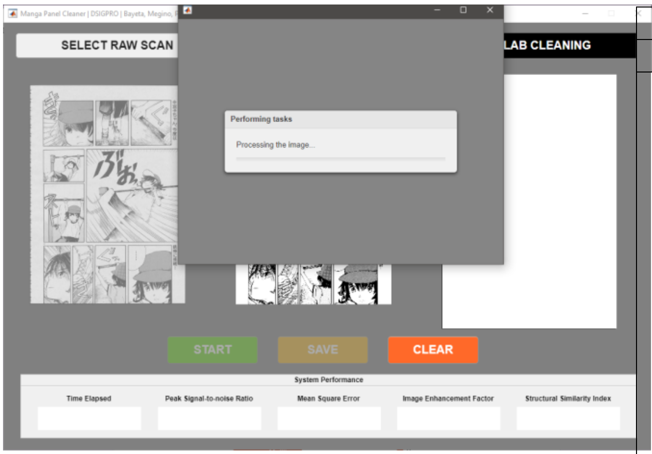



Fig. 5. MPC Program while processing the raw scan.

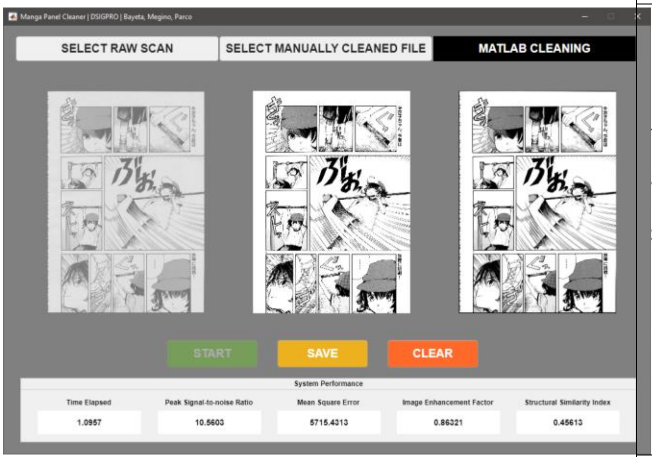



Fig. 6. MPC with the cleaned panel and performance metrics

## IV. Results and Discussion

### A. System Outputs

This section highlights the results of putting four raw manga scans through the system. Table I shows each raw scan compared to both the manually cleaned files and its cleaned or processed versions.

TABLE I. System Outputs

| Manga Panels | |
|---|---|
| ***Manually Cleaned*** [15] | ***Cleaned Image*** |
|  |  |
|  |  |
|  |  |
|  |  |

### B. System Performance

TABLE II. System Performance

| Panel Number | Performance Metrics | | | | |
|---|---|---|---|---|---|
| | *Time Elapsed (s)*[a] | *Peak Signal-to-noise Ratio (dB)* | *Mean Square Error* | *Image Enhancement Factor* | *Strucutal Similarity Index* |
| 1 | 1.0517 | 10.3542 | 5993.2037 | 1.0061 | 0.45165 |
| 2 | 0.8951 | 10.5603 | 5715.4313 | 0.8632 | 0.45613 |
| 3 | 0.9043 | 10.9122 | 5270.5959 | 0.9000 | 0.44358 |
| 4 | 0.8852 | 11.6655 | 4421.28 | 1.0971 | 0.51914 |

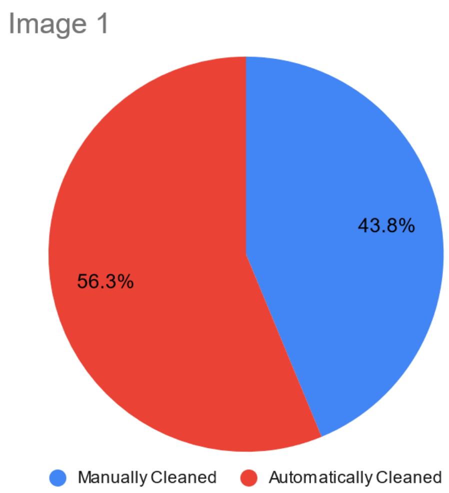
a. Varies with computer performance. Measured on a PC with Ryzen 7 2700, 32 GB RAM

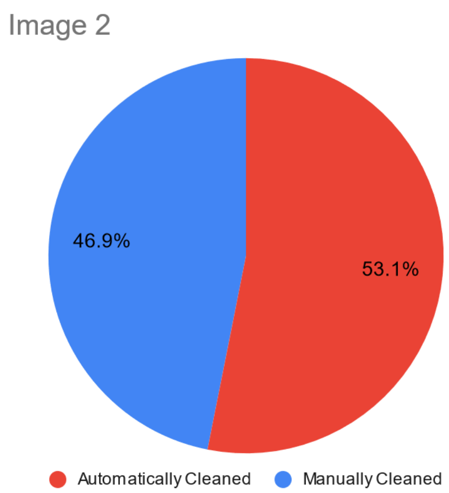
Table II shows the performance metrics result of MPC. The Peak Signal-to-noise Ratio between the manually cleaned files and the clean panel ranges between 10.3542 in the first panel and 11.6655 in the fourth panel.

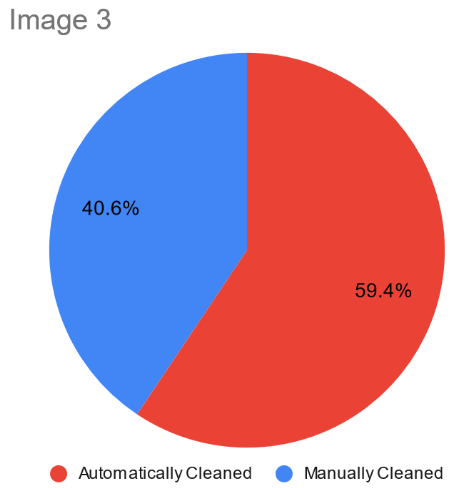
Out of the four processed images, only images 1 and 4 showed an image enhancement factor above 1.0. Despite this, the lowest Image Enhancement factor reported was panel number 2 with 0.8632. The results of the Structural Similarity Index were found to correlate most with the Image Enhancement Factor, with a correlation coefficient of 0.8040. This is in contrast to its correlation with PSNR (0.6146) and its correlation with MSE (-0.5871). From this, it could be inferred that the similarity of a processed image to the original image, in this case, the processed panel to the manually cleaned files, would have an effect on the predicted perceived quality of the image.

Additionally, the Image Enhancement Factor indicates that the predicted perceived quality of the cleaned frame is at least 86.32% of the perceived quality of the manually cleaned files.

Finally, the elapsed time shows that on a performant computer, the time of completion for a frame to be cleaned can vary between 1.0517 and 0.8852 seconds. Over 30 panels can be cleaned within a minute.



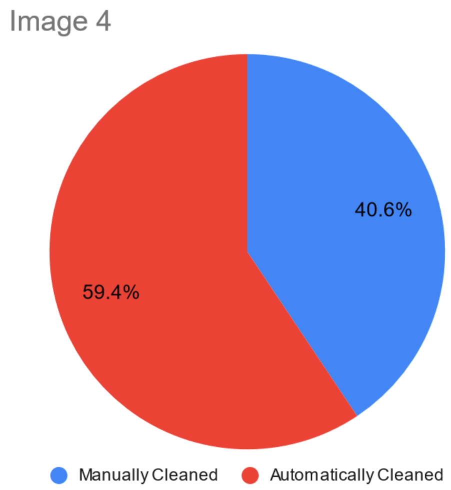




Fig. 7. Perceived quality survey results

Thirty-two (32) survey participants were chosen at random from a pool of manga readers. This was done to get relevant data on perceived quality from people who frequently read digital manga and who might be especially sensitive to the difference between the two manga frames.

Figure 7 shows that between the manually cleaned and the automatically cleaned panels, the sample slightly favored the automatically cleaned panels. However, for all frames, at least 40% of participants still found the manually cleaned panels to be of better quality.

These findings are compliant with our findings from the quantitative metrics for performance measurement. The perceived quality between the manually cleaned images and the automatically cleaned panels is similar or comparable to one another.

## V. Conclusion and Future Improvement

This study illustrated the development and implementation of MATLAB App Designer as a GUI platform for Manga Panel Cleaner, an automated manga cleaning system. The system was able to show comparable results to manually cleaned files and manually cleaned manga panels in an automated manner, allowing the process to be completed in seconds. Moreover, the survey results reinforced the comparable quality between the manually cleaned images and the automatically processed images. For future developments, the researchers will look into batch processing to provide a better and faster experience for users. Additionally, they will look into more image processing techniques to attempt to improve the measured performance metrics for better quality images.

## Acknowledgment

We would like to thank Dr. Gerald P. Arada and Dr. Carlo Noel E. Ochotorena for their teachings and guidance in our Digital Signal Processing Lecture (DSIGPRO) and Laboratory (LBYCPA4) courses, respectively.

## References

[1] A. Hashimoto, "Popular Culture: Manga | Japan Module." https://www.japanpitt.pitt.edu/essays-and-articles/culture/popular-culture-manga (accessed Jan. 04, 2022).

[2] mwsmws22, "How did I do cleaning? (Tsurebito) | MangaHelpers," 2016. https://mangahelpers.com/forum/threads/how-did-i-do-cleaning-tsurebito.3006233/ (accessed Jan. 04, 2022).

[3] D. Li, S. Simske, and R. M. Mersereau, "Image denoising through support vector regression," *Proc. - Int. Conf. Image Process. ICIP*, vol. 4, 2007, doi: 10.1109/ICIP.2007.4380045.

[4] V. Karnati, M. Uliyar, and S. Dey, "Fast non-local algorithm for image denoising," *Proc. - Int. Conf. Image Process. ICIP*, pp. 3873–3876, 2009, doi: 10.1109/ICIP.2009.5414044.

[5] M. Liu and Y. Wei, "Image Denoising Using Graph-Based Frequency Domain Low-pass Filtering," *2019 IEEE 4th Int. Conf. Image, Vis. Comput. ICIVC 2019*, pp. 118–122, Jul. 2019, doi: 10.1109/ICIVC47709.2019.8980994.

[6] X. Huai, K. Lee, and C. Kim, "Image noise reduction by dynamic thresholding of correlated wavelet intensity and anisotropy," *2010 Int. SoC Des. Conf. ISOCC 2010*, pp. 28–33, 2010, doi: 10.1109/SOCDC.2010.5682982.

[7] X. B. Wang, "Image enhancement based on lifting wavelet transform," *Proc. 2009 4th Int. Conf. Comput. Sci. Educ. ICCSE 2009*, pp. 739–741, 2009, doi: 10.1109/ICCSE.2009.5228128.

[8] A. Nasonov and A. Krylov, "Image sharpening by grid warping with curvature analysis," *Proc. - 15th Int. Conf. Signal Image Technol. Internet Based Syst. SISITS 2019*, pp. 262–267, Nov. 2019, doi: 10.1109/SITIS.2019.00051.

[9] J. Panda and S. Meher, "An Efficient Image Interpolation Using Edge-error Based Sharpening," *2020 IEEE 17th India Counc. Int. Conf. INDICON 2020*, Dec. 2020, doi: 10.1109/INDICON49873.2020.9342212.

[10] A. Alsam, I. Farup, and H. J. Rivertz, "Iterative sharpening for image contrast enhancement," *2015 Colour Vis. Comput. Symp. CVCS 2015*, Sep. 2015, doi: 10.1109/CVCS.2015.7274891.

[11] Y. Raphiphan, S. Wattanakaroon, and S. Khetkeeree, "Adaptive High Boost Filtering for Increasing Grayscale and Color Image Details," *Proc. 2020 Int. Conf. Power, Energy Innov. ICPEI 2020*, pp. 69–72, Oct. 2020, doi: 10.1109/ICPEI49860.2020.9431572.

[12] E. Erwin, A. Nevriyanto, and D. Purnamasari, "Image enhancement using the image sharpening, contrast enhancement, and Standard Median Filter (Noise Removal) with pixel-based and human visual system-based measurements," *ICECOS 2017 - Proceeding 2017 Int. Conf. Electr. Eng. Comput. Sci. Sustain. Cult. Herit. Towar. Smart Environ. Better Futur.*, pp. 114–119, 2017, doi: 10.1109/ICECOS.2017.8167116.

[13] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising," *IEEE Trans. Image Process.*, vol. 26, no. 7, pp. 3142–3155, Jul. 2017, doi: 10.1109/TIP.2017.2662206.

[14] "2 Jikan me Hoken Taiiku - Matsuyama Saya | DeadScanlations." https://deadscanlations.tumblr.com/tagged/2 Jikan me Hoken Taiiku (accessed Jan. 04, 2022).

[15] S. Matsuyama, "2nd Period Health & PE," 2009. https://mangakatana.com/manga/2nd-period-health-pe.18600 (accessed Jan. 04, 2022).