

# Image Processing with Virtual Pencil and Virtual Mouse using Webcam Recognition for Drawing

Rikip Ginanjar<sup>1</sup>, Sijoatmodjo Ivan Carnelius<sup>2</sup>  
<sup>1,2</sup>Faculty of computing, President University  
Jl. Ki Hajar Dewantara, Cikarang Baru, Bekasi, 17550  
<sup>1</sup>rikipg@gmail.com  
<sup>2</sup>carnelius.ivan@gmail.com

**Abstract**— Recently, people use physical mouse to control the computer. Virtual Vpaint can be a new innovation to change the physical mouse to virtual mouse, then the users can minimize the cost of hardware consumption and damages. Having virtual mouse, users can control the computer only through webcam to do a lot of things such as drawing, editing images, mouse control, and others. Virtual VPaint application will discuss the development of Computer Vision for drawing an object in Windows form application by using virtual mouse. This application can perform common image editor such as blurring, gray scaling, and negating images. Users can control mouse pointer through the webcam by using object mover which has specific color, then can draw an object on canvas by using virtual pencil.

**Keywords**— image processing, virtual pencil, virtual mouse, webcam recognition, drawing

## I. INTRODUCTION

Computer Vision (OpenCV) was launched in 1999 as an initially Intel Research initiative to advance CPU-intensive applications and optimized by experts in Intel Russia as well as Intel's performance Library Team to develop multi languages toward Operating systems in Computer.

Nowadays, the implementation of Computer Vision such as Virtual Mouse and Pencil concepts are still disguised to human lifestyle because users are used to work with physical hardware such as Mouse. The new innovation of Virtual Mouse and Pencil give breakthrough to human lifestyle. Virtual Mouse and Pencil supplant the physical mouse usage, users only need to move their hand through its direction towards screen and its more simple compare with physical Mouse. By using the Human Computer Interaction knowledge, Computer Vision-based gesture and object recognition, the implementation of Virtual Mouse and Pencil can be applied through Notebook.

There are many image editor applications in windows application. However, the implementation of virtual mouse towards image editor application is still least. The objectives of this research:

- To create and implement Virtual Mouse development into Desktop application towards Webcam
- To create and implement Virtual Pencil development into Desktop application towards Webcam
- To create and implement Convolution concept in image editor using Virtual Mouse

Virtual Vpaint application has some limitation such as only recognition red and yellow color, so then the user needs specific color property when running the application. Besides that, users need to wait 3 seconds when performing left click. The background of webcam and lightning room should not be in bright color. The data type of open and save image is bitmap while virtual mouse only performs left click feature.

## II. METHODOLOGY

### II.1 Gray Scale

Common strategy in gray scaling image, programmers use the principles of photometry or colorimetric to match the luminance of grayscale image to luminance of the original color image. Photometry is a science of the measurement of light, in terms of its perceived brightness to human's eye while colorimetric is the science and technology to quantify and describe physically the human color perception. It uses the weighted sums of RGB color model to grayscale representation of its luminance. The expression for those theory as:

$$Y' = 0.2126R' + 0.7152G' + 0.0722B'$$

### II.2 Negative Image

Normal images known as Positive Images are images where people are used to see in ordinary image, Negative image is color which change 180 degrees from the origin as:

$$R'=255-R; G'=255-G; B'=255-B;$$

### II.3 Gaussian and Convolution Function

Mathematically, Gaussian Smoothing assembling can be implemented by convolving the Gaussian Function. Convolution is a mathematic operation which can be implemented in digital image processing such as blurring and sharpening. Calculation process is used a sliding window called convolution matrix or kernel, it is matrix m x n which position can be moved and iterated toward image coordinate which is shown in Figure 1.

	1	4	7	4	1
	4	16	26	16	4
$\frac{1}{273}$	7	26	41	26	7
	4	16	26	16	4
	1	4	7	4	1

Figure 1. Gaussian Function

The value of Gaussian function is shown by matrix 5x5 below where this value will be used as kernel which is shown in Figure 2.

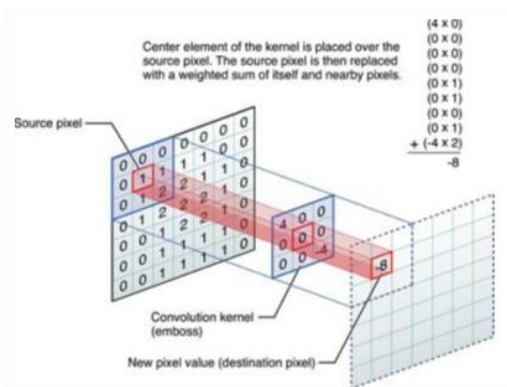


Figure 2. Convolution Process

There is a special treatment, crop method, for every edge pixel from the image, the value will be passed to result image to avoid stack overflow when performing convolution process. The underline pixel which shown in Figure 3 is the edge of image.

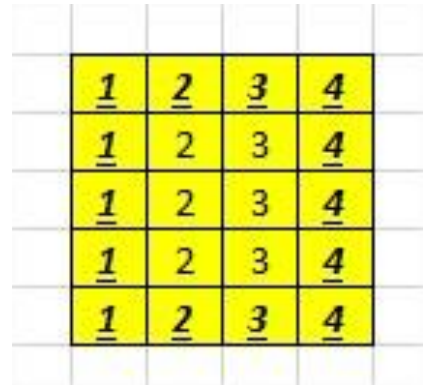


Figure 3. Crop Method

### II.4 Optimization Arithmetic Mean

The Arithmetic mean, known as average is the most commonly used and readily understood measure of central tendency. In statistics, the term average refers to any of the measures of central tendency. The arithmetic mean defines as being equal to the sum of the numerical values of each and every observation divided by the total number of observations. The calculated value is the coordinate input from the user using webcam recognition which can be seen in Figure 4.

$$A = \frac{1}{n} \sum_{i=1}^n a_i.$$

Figure 4. Arithmetic Mean

### II.5 CIRCLE HOUGH BOUNDARY

Circle Boundary is an algorithm to detect circle object and optimize it although the captured object is not in 100% circle because of some noises. The algorithm will calculate and extract the full data of circle such as in Figure 5. There are some part of ball hidden by hand when user hold the yellow ball, however the program using circle boundary recognition will extract the feature and get optimized result (red circle) which can be seen in Figure 5.



Figure 5: Circle Hough Detection

### III. RESULTS

In order to evaluate the effectiveness of the proposed method in previous section, there is some implementation testing such as,

TABLE 3.1.  
VIRTUAL MOUSE EVALUATION

No	Scenario	Expected Result
1	Navigate the cursor	System gets coordinate when webcam succeed to catch an object then continue the process mouse navigation. label text notification "in control", and cursor moves
2	Perform left click	System does not detect an object color in certain time, left click label text notification "true", performs left click.
3	New Navigation coordinate	System does not detect object color in certain time, mouse status label text notification "mouse move", system detects new coordinate point to navigate the mouse

Table 3.1 represents the testing result for virtual mouse function such as navigate, left click, new navigation point towards the desktop.

TABLE 3.2.  
VIRTUAL PENCIL EVALUATION

No	Scenario	Expected Result
1	Choose color	System will change the pencil color based on what color button is chosen by user in field color list, the color can be seen when user starts to draw anything in picture box
2	Erase object	When eraser button is clicked, system will change the color to white same with background and the range of eraser will be larger than pencil
3	Pencil Status	System will change the label text "true" or "false" when the system does not detect an object mover in certain time
4	Pencil activation button	System will change the pencil activation button text "true" or "false" when system performs left click towards button
5	Draw an object	System will draw an object into picture box 1 with specific color when the pencil status is true and pencil activation button is turn on, system cannot draw an object if both conditions are not fulfilled

Table 3.2 represents the testing result for virtual pencil features in application where users can choose the color, erase and draw an object in provided canvas.

TABLE 3.3.  
OBJECT RECOGNIZER EVALUATION

No	Scenario	Expected Result
1	RGB color recognizer setting	System will change the color threshold recognizer based on track bar value which user chooses
2	Detect Coordinate - Camera Projection	System will show the result image in white and black color based on color threshold recognizer value, system will detect center coordinate of white color object in projection, system will not send any values if there is no white color in projection.
3	Detect Multiple Coordinate	If there are more than one object can be detected by system, system will read all coordinates alternately
3	Virtual Mouse activation button	System will turn off the virtual mouse functionality when the checkbox is checked
4	Webcam is not available	When system cannot detect the webcam functionality, system will give a popup message and system will terminate process and close the application

Table 3.3 represents the testing result for webcam recognition towards color and send the coordinate data to the system.

Table 3.4 represents the testing result for image editor features such as smoothing, gray scaling and negating the images. System will read original image and produce result image after editing the image using provided function.

TABLE 3.4.  
IMAGE EDITOR EVALUATION

No	Scenario	Expected Result
1	Smoothing image	System will perform smoothing towards image function when menu smoothing is clicked by user, then display the result image into picture box 2. If there is no image loaded then the function does not work and nothing happens.
2	Gray scaling image	System will perform gray scaling towards image function when menu gray scaling is clicked by user, then display the result image into picture box 2. If there is no image loaded then the function does not work and nothing happens.
3	Negative image	System will perform negative image towards image function when menu negative image is clicked by user, then display the result image into picture box 2. If there is no image loaded then the function does not work and nothing happens.

TABLE 3.5  
DISPLAY EVALUATION

No	Scenario	Expected Result
1	Display Image	System will display the result image after the system performs image editor function based on what user chooses, the results will be displayed in right-side Picture box
2	Undo Image	System will display previous image which displayed in right-side picture box. If there is no image loaded when performing undo, there will be no action when user performs the features

Table 3.5 represents the testing result for display result image into canvas after doing image editor.

TABLE 3.6.  
SAVE AND LOAD EVALUATION

No	Scenario	Expected Result
1	Save Image	System will save image from all picture boxes into specific folder. The data type is in Bitmap. If there is no image to be saved, the system will automatically declare the picture box as white color for all pixel same as background then saved.
2	Load Image	System will load only for Bitmap file then display the images into picture box 2. The displayed image size will be stretched with picture box size temporarily. However, when user saves the image, the original size will be back on.

Table 3.6 represents the testing result for save and load images between specific folder and application in bitmap data type.

#### IV. DISCUSSION

In this section, there are some discussion why Virtual Vpaint uses explained method, they are,

- Optimization Arithmetic Mean

Virtual Vpaint uses Arithmetic Mean algorithm where system collect several coordinate and do average calculation for x and y coordinates. For the initial case, system will detect 7 coordinates firstly then send optimized coordinate to the system. Later on, system will exclude the first coordinate which is caught before and move the array to N-1 and fill the latest array with newest data input from user. The advantages of this method is: System can reduce the noisy movement when there is a little bit movement such as (x+1), (x-1), (y+1), (y-1) where x and y are the coordinates.

- Convolution for Image Editor

Convolution is a concept to manipulate the RGB color of related pixel by some set kernel value. The system will edit pixel by calculating not only the specific pixel but also the value around the specific pixel, so then the result pixel can be optimized because the value already integrated with surrounded pixel. The advantage of this method is: System can do smoothing perform well and have good color combination because the value of each single pixel already integrated with surrounded pixel.

- Circle Hough Detection

Circle Hough Detection is a concept where system has a function to detect a circle from specific RGB Color. The images comes from webcam, so the quality of webcam affects the Circle Hough performance. The advantages of this method are:

- Users do not need to take a picture of every edge of image and store to XML file in order to make system can recognize the object mover.
- System can detect whole object although half of object is hidden or not caught by webcam, especially for circle which can be seen in Figure 5.

#### V. CONCLUSIONS

Virtual VPaint can implement object detection using object circle boundary. Mean algorithm works well to optimize the movement when controlling mouse pointer and drawing an object. The environment such as lighting, object color are really affect the webcam recognition performance. Convolution, color complement diagram, and RGB Cartesian Coordinate System theorem can be used to create functions for image editor such as Smoothing, Gray scaling, and Negating images.

In the future, Virtual Vpaint will use training detection function, more optimization method and advance hand recognition so then the program can perform better and faster in recognizing the object.

#### REFERENCES

- [1] Advantages and Disadvantages RAD (Accessed 7 October 2015) <http://www.my-project-management-expert.com/the-advantages-and-disadvantages-of-rad-software-development.html>
- [2] Arithmetic Mean. Jose, D. Perezgonzales. Arithmetic Mean, 2012
- [3] Blur Image (Accessed 10 October 2015) <https://futurestud.io/.../how-to-blur-images-efficiently-with-androids-rem>
- [4] Chart windows usage (Accessed 7 October 2015) <https://www.netmarketshare.com/operating-system-market-share.aspx?qprid=10&qpcustomid=0&qptimeframe=Y>
- [5] Computer Definition Concept. Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. SURF: Speeded up robust features. In European Conference on Computer Vision, 2006
- [6] Computer Vision CS291A00. "Computer Vision I", University of California, San Diego, 2004
- [7] Computer Vision Definition. (Accessed 9 October 2015) <http://programming.computervision.com/>
- [8] Convolution. Wendy, Manipulasi Citra Wajah pada Webcam untuk Aplikasi Skype, 2014
- [9] Convolution and Image Processing (Accessed 9 October 2015) Hirschman, Isidore and Widder, David Vernon. The Convolution Transform, 2012
- [10] Convolution Kernel (Accessed 10 October 2015) <http://dev.theomader.com/gaussian-kernel-calculator/>
- [11] First Mouse (Accessed 7 October 2015) [https://en.wikipedia.org/wiki/Mouse\\_\(computing\)](https://en.wikipedia.org/wiki/Mouse_(computing))
- [12] First Webcam (Accessed 7 October 2015) [https://en.wikipedia.org/wiki/Trojan\\_Room\\_coffee\\_pot](https://en.wikipedia.org/wiki/Trojan_Room_coffee_pot)
- [13] Grayscale (Accessed 9 October 2015) <https://en.wikipedia.org/wiki/Grayscale>
- [14] Human Computer Interaction (Accessed 9 October 2015) <https://www.interaction-design.org/literature/book/the-encyclopedia-of-human-computer-interaction-2nd-ed/human-computer-interaction-brief-intro>
- [15] Kinect (Accessed 10 October 2015) <https://dev.windows.com/en-us/kinect>
- [16] Leap Motion (Accessed 10 October 2015) [www.engadget.com/2013/07/22/leap-motion-controller-review/](http://www.engadget.com/2013/07/22/leap-motion-controller-review/)
- [17] Negative Image (Accessed 10 October 2015) [www.imageprocessingplace.com/DIPUM2E\\_Chapter03\\_Pgs\\_80-114\\_](http://www.imageprocessingplace.com/DIPUM2E_Chapter03_Pgs_80-114_)
- [18] Object Recognition Process (Accessed 9 October 2015) Thomas B. Moeslund and Erik Granum, "A Survey of Computer Vision-Based Human Motion Capture," Elsevier, Computer Vision and Image Understanding, 2001
- [19] OpenCV (Accessed 9 October 2015) [http://www.seas.upenn.edu/~bensapp/opencvdocs/ref/opencvref\\_cv.htm](http://www.seas.upenn.edu/~bensapp/opencvdocs/ref/opencvref_cv.htm)
- [20] OpenCV EmguCV (Accessed 9 October 2015) <http://opencv.org/>
- [21] Relative\_luminance (Accessed 10 October 2015) [https://en.wikipedia.org/wiki/Relative\\_Luminance](https://en.wikipedia.org/wiki/Relative_Luminance)
- [22] Virtual Mouse Concept (Accessed 9 October 2015) <http://classes.engr.oregonstate.edu/eecs/fall2013/ece441/project.php?ID=30>