Robust Color Tracking to Ambient Light Changes

Darío José Mendoza Chipantasi, Nancy Velasco, David Rivas and Víctor H. Andaluz
Departamento de Energía y Mecánica,
Departamento de Ciencias Exactas,
Departamento de Eléctrica y Electrónica,
Universidad de las Fuerzas Armadas ESPE, Sangolqui, Ecuador.
djmendoza@espe.edu.ec

***Abstract*—The use of color is a good technique for tracking objects. This technique has offered many advantages related to the robustness over occlusions, rotations, different geometries and scales. However, there are many studies that claimed that this technique generates excessive computational cost, and loss of reference due to the similar background color or loss of reference due to drastic changes in lighting. Therefore, there is a need to create a robust algorithm to prevent all the disadvantages mentioned above. This algorithm was mainly built on the analysis of the hue and saturation channels in the histogram based on the color space (hue, saturation and brightness - HSV). To resolve the disadvantages, an automatic recalculation of the histogram at each n frames has been developed. The algorithm has been enhanced to make it able to do all the process in real-time in large images up 1920x1080 pixels. The possible use of the algorithm is for controlling an automatic unmanned vehicle that will track a person.**

***Index Terms*—Robust Tracking; Histogram Recalculation; Color Tracking.**

I. INTRODUCTION

Tracking objects on the images composed by video is a common topic in computer vision, and there are many potential applications to facilitate this task. Analysis used in object tracking is mainly focused on the recognition of specific characteristics, for example identifying shape, texture and color of target objects.

Color information is an important key in object tracking. Analyzing the histogram is a useful key because color is invariant to the position and pose of the objects. A study in [1] investigates the contribution of color in a tracking-by-detection framework, but one of the conclusions is that analyzing each frame to find the color reference causes a lot of computational cost, slowing down the processing and affecting the real time tracking. Focusing on this issue, our work tries to improve the performance for visual tracking using color attributes.

Some developments have been previously done, such as [2] that presents a tracking of the target using color-based particle filter that uses the analysis of particle filter to identify a point mass in the target figure and later perform object Sequential movement estimation. The aim of the study was to find and represent the object by its dominant color and analyze the histogram for optimum tracking objects algorithm. The study concludes that it is difficult to be precise and remain stable in a complex environment. Another type of method [3] adopted the Gray World Assumption (GWA) where it emulated the color constancy in human vision. The average color in the vision is viewable as a neutral gray. For this reason, based on based on the GWA, the object color is defined as relative values from the average color in the frame of the image sequence. In [4], Ji Tao presented a complete automated camera-based monitoring system that makes use of low level color features to perform detection, tracking and recognition of multiple people in video sequences. In [5], the authors proposed a "color tracking method mean-shift" [16] based under luminance change. They proposed an algorithm that can calculate a simple color for processing, and this method is faster than the conventional normalized vector distance (NVD). A study reported in [6] introduced a multi-part color modeling to capture a rough spatial layout ignored by global histograms. It is an incorporation of a relevant background color model and extended to multiple objects. Pfinder models pixel color variation using multivariate Gaussian for people tracking was developed in [7]. Kwolek [8] applied color, stereovision and elliptical shape features of faces and established a system that robustly tracks and detects faces. McKenna and Elgammal [9] both employed color models of people for occlusion reasoning: They attempted to segment people in an occlusion by deciding the origin of a person (or color model) from each pixel. As color models in [10, 11] and gray-level templates in [12], the reference can be extracted from the first frame and kept it during all the process. Odisio et al. [13] used shapes and appearance models to track talking faces.

A study in [14] demonstrated the use of multiple Instance Learning (MIL) rather than traditional supervised learning to avoid problems and produce a more robust tracker with fewer parameters for checking. The conclusion of this work is to create a novel online MIL algorithm to perform an object tracking that achieves better results when working in real-time.

We focus on a robust color tracking based on [15] of Bradski, who developed a continuously adaptive mean shift algorithm to track human faces. We tested it to strong lighting changes by recalculating the histogram every n frames.

This paper is organized in sections: Section II explains the use of images by summarizing the steps of the algorithm. In Section III, the results of the experimental work are presented, and finally, the conclusion of this paper is presented in Section IV.

II. EXPERIMENT

The algorithm is divided into several phases, which are concatenated consecutively to achieve the objectives of the project. Figure 1 illustrates the phases in a block diagram.

A. Select the starting color

Firstly, a function is created to identify the initial rectangular tracking area with the first and second click on the 'mouse'. It is useful to make the initial sample of the color region which will be analyzed by the histogram. As previously explained, this algorithm is based on obtaining the color sample of a given region because this will be the basis of robust tracking along the algorithm.

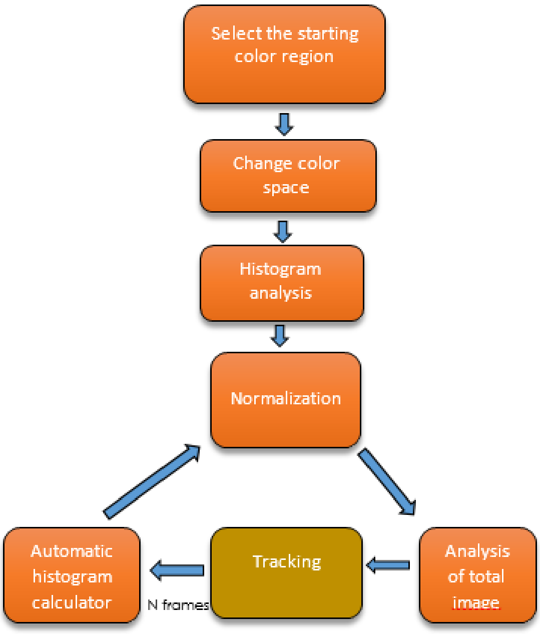


Figure 1: Block diagram of the processes of the system

B. Change color space

The next step is to initialize the camera because it will be the source of frames acquisition, although the algorithm can accept a previously stored video as a source for analysis. The images usually captured by the cameras are in Red, Green, Blue (RGB) color space, but from previous experience, it is necessary to change it to Hue, Saturation, Value (HSV) color space for more robust monitoring. This is because the primary advantage is the isolation the brightness into a single channel (Value), leaving only the hue (tint) and the saturation channels for the analysis

C. Histogram analysis

When the image is in HSV, a function CALCHIST is used to analyze the distribution of color within an image. This method helps to recognize the area for tracking in each frame analyzed and create the HISTOGRAM to analyze each one of the surface represented by each bar, where each bin is proportional at the frequency of the identified values.

In this step, the core of the algorithm is realized. However, a lot of continuous processing is needed to make it a robust color tracking to ambient light changes. CalcHist command is necessary for the histogram analysis as this method analyzed the histogram within hue and saturation channels. The calculation operation is performed only in the selected region to extract the specific characteristics (previously selected with the mouse).

D. Normalization

The aim is to normalizr these values within the range from 0 to 255 based on the minimum and maximum values obtained by the histogram. This step is necessary because it changes the scale of calculating histogram and redistributes all the information to perform a better color categorization.

E. Analysis the total image

The back-projection histogram calculation is necessary for image analysis, where the work of back projection is similar to the work with CalcHist. In this work, the content of each pixel is evaluated based on the same color of the sample. If it meets the condition, the pixel is stored within the corresponding bin. The function creates a new image where the pixels that have the same color as the sample are represented with white color and the leftover pixels with black color, resulting a binary image where only the areas that contain the color tracking appear.

In terms of statistics, the function calculates the probability that each value analyzed corresponds with the empirical distribution to the histogram in search.

When the color sample references into the frame are not compatibility, the program automatically terminates the process and delivers an error message for user, explaining that it is not possible to find a color for tracking or the sample is too small.

F. Tracking

We used the information provided by calBackProject for doing a tracking because it shows only the regions of color that have the same characteristics as the sample. Camshift command calculates the probability of the color sample found by calcBackProject that corresponds to the initial reference tracking because it can appear in other regions in the image that contains the same color characteristics, but it does not correspond to the object tracked. Besides, when using the Camshift command to complete the task, the algorithm marks the area that contains the reference, providing an augmented reality and feedback about what color region is being followed by the algorithm.

Mathematically, the function "Camshift" increases the gradient of a probability distribution "backprojected" calculated from a histogram rescaling of color to find the nearest peak within a search window. With this, the average location of the target is derived by calculating the moment in image the zero (Equation (1)), first (Equation (2) and Equation (3)) and second (Equation (4) and Equation (5)).

$$M_{00} = \sum_x \sum_y P(x,y) \tag{1}$$

$$M_{10} = \sum_x \sum_y xP(x,y); \tag{2}$$

$$M_{01} = \sum_x \sum_y yP(x,y) \tag{3}$$

$$M_{20} = \sum_x \sum_y x^2P(x,y); \tag{4}$$

$$M_{02} = \sum_x \sum_y y^2P(x,y) \tag{5}$$

where ***P(x,y) = h(I(x,y))*** is the probability of the position and distribution ***(x,y)*** into within the search window ***I(x,y)*** that is calculated from the histogram ***h*** of ***I***. The average position of the search object can be calculated with Equation (6) – (8):

$$x_c = \frac{M_{10}}{M_{00}}; \tag{6}$$

$$y_c = \frac{M_{01}}{M_{00}} \tag{7}$$

$$ratio = \frac{\frac{M_{20}}{x_c^2}}{\frac{M_{02}}{y_c^2}} \quad (8)$$

while its aspect ratio is defined by Equation (9) and Equation (10):

$$width = 2M_{00} * ratio; \quad (9)$$
$$height = \frac{2M_{00}}{ratio} \quad (10)$$

G. Automatic histogram calculator

In order to make a robust tracking, a function that automatically calculates the histogram at every n frames was implemented. This function helps to obtain a new histogram to track the color even if the lightning conditions change fast.

Currently, the program recalculates the histogram at every n frames. However, if the change of color reference is sudden, the algorithm automatically calculates the histogram of another color. For this reason, a future work will be to make a comparison of the recalculated histograms, and if the information is not similar, the program should stop following the object.

The program has a limitation of minimum size for tracking, thus it ends the monitoring when the reference is very small.

The user interface is friendly. This program could be the base for implementing an autonomous unmanned vehicle that mobilizes large loads without the need for effort.

III. Results And Discussion

The challenge was to develop an algorithm for doing a robust color tracking. It can track the color even if the lighting conditions are changing. The algorithm was developed using OpenCV library and C++ programing language. Another functionality is the computational cost of the computer that is set as low as possible.

Figure 2 shows: a) region of interest (ROI) selected by the user. This is very useful because the program could follow any color under the user's request. If the object to follow moves too much, a small functionality was implemented to stop the image acquisition (pause function by pressing the P key on the keyboard) until the user can select the area of the object. b) The original image is changed from RGB to HSV color space to allow for better analysis.

We used only Hue and Saturation channels to avoid the influence of brightness on image because the channel Value contains the information about the brightness in the image. Although the algorithm is robust, we may have trouble when trying to follow the reflecting colors even in the sunniest days because these colors are captured by the webcam as if they were white, disabling the proper functioning. c) calculate the histogram, normalize it and analyze the whole image looking for a region that has the same characteristics of the original color. We analyzed only the ROI to reduce the processing time image and achieved real time processing; d) a mask was created: It was a white rectangle in the black image, e) the algorithm enclosed the tracking area, this gave a kind of augmented reality for the user. Finally, Figure 3 shows how the algorithm addressed to the external changes of ambient light that was produced by a flashlight focused on the color region or focused on the camera, proving the robust tracking color.

IV. Conclusion

The implemented algorithm was capable to maintain a tracked color even if the light conditions change quickly. Another advantage is that it is possible to perform a color tracking although the background of the image has a similar color, as can be seen in Figure 4(a) and 4(b). However, an inconvenient of this approach is that the processing of all the information content in a full image caused a slow processing when the background has the same color than the sample. This was another challenge because the actual algorithm is capable to analyze images in full high definition (1920x1080) pixels in real time.

To test the algorithm, we used different kinds of objects and with different colors. We also added other window that shows the histogram in real time. The windows that contain histogram information are used only for demonstrating the changes that was produced when the light conditions varies, thus obtaining a robust tracking. There exists a lot of possible applications of algorithm: It can track a person or other moving objects depending on the color, allowing a autonomous navigation in unmanned vehicles.

After conducting many tests, the algorithm proves to be very strong for ambient light changes. However, we recommended that more analysis for comparing histograms should be done when a new histogram was selected because the fast movements of the object can cause changes in tracking the color at the time of color reference recalculation. This project can be implemented to control an air or land unmanned autonomous vehicle.

Developing an algorithm that combines robustness, but without creating an excessive burden of processing the computer has a nontrivial challenge.

Acknowledgement

We are grateful for Mr. Sergio Domínguez and Mr. Adrián Carrió for guiding us in learning computer vision during the studies in the “Universidad Politécnica de Madrid”. Besides, our sincere thanks for all the support received from our professors, friends and department managers: Energy and Mechanics (Mr. Marco Singaña), Exact Sciences (Ms. Jacqueline Pozo) and especially the team that makes up the Mechatronics Engineering directed by Mr. Vicente Hallo, friend and career director; who belong to the “Universidad de las Fuerzas Armadas - ESPE Extension Latacunga”.

References

[1] Danelljan, M., Khan, F., Felsberg, M., & Weijer, J. 2014. Adaptive Color Attributes for Real Time Visual Tracking. *IEEE Conference on Computer Vision and Pattern Recognition*. 1090-1097.

[2] Mukhtar, A., Xia, L. 2014. Target Tracking Using Color Based Particle Filter. *2014 5th International Conference on IEEE En Intelligent and Advanced Systems, (ICIAS*). 1-6.

[3] Matsumoto, T., Kimachi, A., Nishi, S., & Ikoma, N. 2014. Robust Color Objects Tracking Method Against Illumination Color Change., *2014 Joint 7th International Conference on Soft Computing and Intelligent Systems (SCIS) and IEEE 15th International Symposium on Advanced Intelligent Systems (ISIS).* 718-722.

[4] Tao, J., Tan, Y. P., Lu, W. 2007. Robust Color Object Tracking with Application to People Monitoring. *International Journal of Image and Graphics.* 7(02):227-254.

[5] Hidaka, K. 2009. Robust Color Tracking Based On Mean-Shift Under Illuminance Change. *ICCAS-SICE 2009*.1827-1830.

[6] Perez, P., Hue, C., Vermaak J. and Gangnet M.. 2002. Color Based Probabilistic Tracking, *ECCV*. 1:661-675.

[7] Wren, C., Azarbayejani, A., Darrell, T., & Pentland, A. 1997. Pfinder: Real-Time Tracking of the Human Bod. in *IEEE Transactions On Pattern Analysis And Machine Intelligence*, 19(7):780-785.

[8] Bogdan, K., 2003. Object Tracking System Based on Color, Stereovision and Elliptical Shape Features, *AVSS*

[9] McKenna, S. J., Jabri, S., Duric, Z., Rosenfeld, A., & Wechsler, H. 2000. Tracking Groups of People. *Computer Vision and Image Understanding*, 80(1):42-56.

[10] Dou, J.-F., and Jian-xun, L. 2014. Robust Visual Tracking Base on Adaptively Multi Feature Fusion and Particle Filter. *Optik International Journal for Light and Electron Optics*. 1680-1686.

[11] Comaniciu, D., & Ramesh, V. 2003. U.S. Patent No. 6,590,999. Washington, DC: U.S. Patent and Trademark Office.

[12] Hager, G. D., & Belhumeur, P. N. 1998. Efficient Region Tracking With Parametric Models Of Geometry And Illumination. *In IEEE transactions on Pattern Analysis and Machine Intelligence*., 20(10):1025-1039.

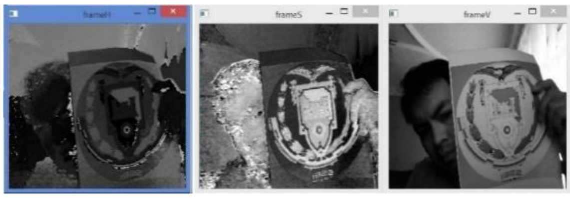
[13] Odisio, M., & Bailly, G. 2003. Shape and Appearance Models Of Talking Faces For Model-Based Tracking. En Analysis and Modeling of Faces and Gestures, 2003. AMFG 2003. IEEE International Workshop.143-148.

[14] Babenko, B., Yang, M. H., & Belongie, S. 2011. Robust Object Tracking With Online Multiple Instance Learning. IEEE Transactions Pattern Analysis and Machine Intelligence. 33(8):1619-1632.

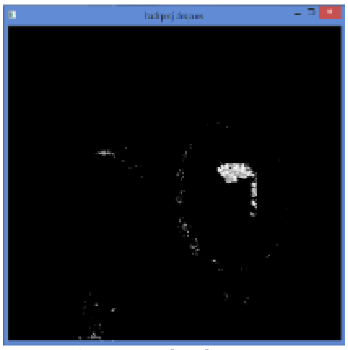
[15] Bradski, G. Real Time Face and Object Tracking As a Component of a Perceptual User Interface. 4th IEEE Workshop Proceedings WACV'98. 214-219

[16] Comaniciu, D., & Ramesh, V. 2003. U.S. Patent No. 6,590,999. Washington, DC: U.S. Patent and Trademark Office.Computer Vision and Pattern Recognition, 2000. Proceedings. *IEEE Conference on, Hilton Head Island*, SC, 2000, 2:142-149.

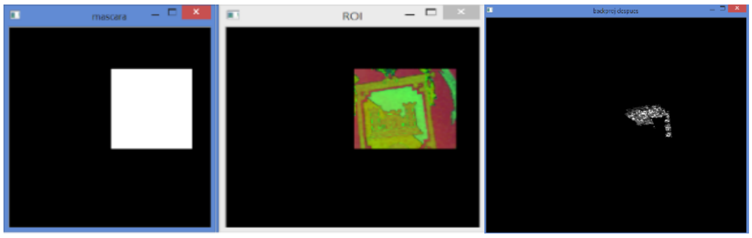
a) ROI selected (pink rectangle



b) images in HSV color space

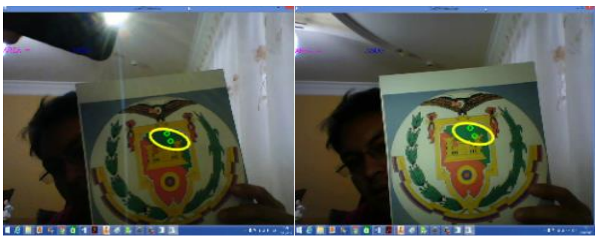


c) Image color detection

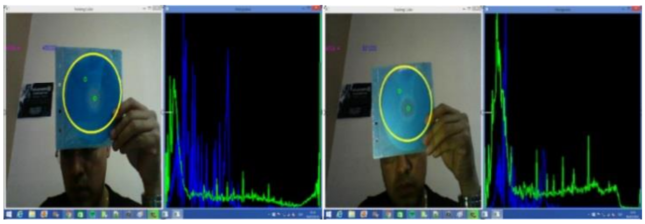


d) ROI to achieve real time processing.

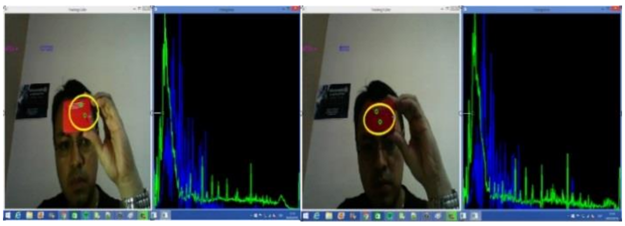
Figure 2: Phases of algorithm



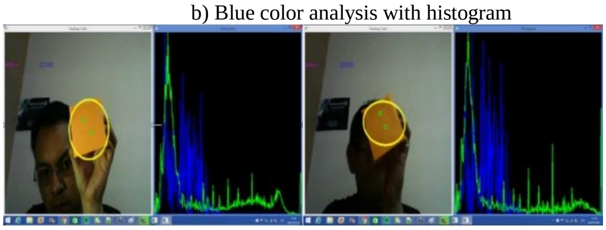
a) Green color analysis without histogram



b) Blue color analysis with histogram

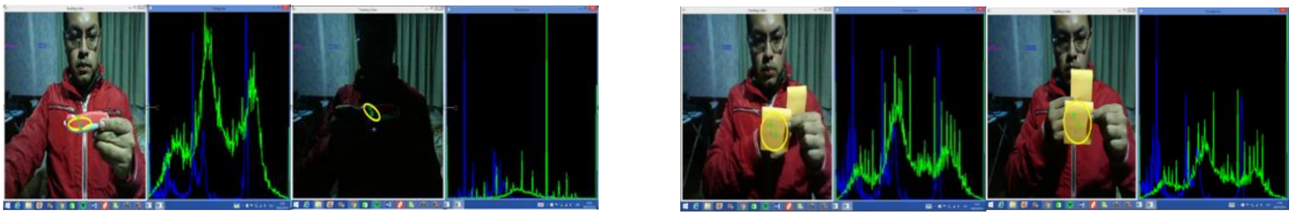


c) Red color analysis with histogram



d) Orange color analysis with histogram

Figure 3: Algorithm response to ambient light changes (blue line corresponds to the behavior of the channel histogram to hue, and green line corresponds to the histogram of saturation channel).



a) Red color tracking with Background of the same color.

b) Orange color tracking with Background of the same color

Figure 4: Algorithm response with background of same color (blue line corresponds to the behavior of the channel histogram to hue, and green line corresponds to the histogram of saturation channel)