

# Robust tracking for augmented reality

José M<sup>a</sup> González-Linares, Nicolás Guil and Julián R. Cózar

Dept. of Computer Architecture, University of Málaga

**Abstract.** In this paper a method for improving a tracking algorithm in an augmented reality application is presented. This method addresses several issues to this particular application, like marker-less tracking and color constancy with low quality cameras, or precise tracking with real-time constraints. Due to size restrictions some of the objects are tracked using color information. To improve the quality of the detection, a color selection scheme is proposed to increase color distance between different objects in the scene. Moreover, a new color constancy method based in a diagonal-offset model and k-means is presented. Finally, some real images are used to show the improvement with this new method.

## 1 Introduction

Augmented reality (AR) applications are gaining popularity due to their huge potential impact on many areas, including medicine, manufacturing, robotics, entertainment, tourism or education. Most AR applications need some enabling technologies including displays, tracking, registration and calibration (Azuma, y otros, 2001). In this work we focus on improving tracking in an AR education application, but our proposal can be easily adapted to other AR applications.

AR education applications allow educators to create a scenario, provide specific information for that scenario, and even embed data seamlessly within the real world context (Bower, Howe, McCredie, Robinson, & Grover, 2013). In this context Seabery, a Spanish technological company, has developed a tool for augmented training for welding called Soldamatic. This application uses a head display and several real objects like welding torches or work pieces to recreate a realistic welding training experience.

One of the most difficult stages in this application is robust and precise tracking of a long rod that simulates the electrode used in the welding process. This rod has to be located as precisely as possible to obtain a good estimation of the welding procedure. Unfortunately, its elongated and thin shape makes very difficult the use of markers that are used in other type of pieces (e.g. probes). Moreover, edges obtained in this small rod are not very precise because the spatial resolution of the camera. Thus, a color based approach has been devised.

This application is used in many different real environments, and the light and working set can vary from one place to another. Differences in illumination cause deviations of the colors measurements that can be solved using a color constancy

algorithm (Buchsbaum, 1980). Most color constancy algorithms are based in some assumptions: usually only a single spatially uniform illuminant is considered, and objects are assumed to be flat, coplanar, and Lambertian. This last assumption doesn't hold in this application because objects are very close to the camera and their angle of view can vary a lot. We have tested several color constancy algorithms with poor results, thus we have developed a new method that obtains better results for this specific problem.

The rest of the paper is organized as follows. A method for selecting colors of the objects in the scene is presented in Section 2. Then, a new algorithm for color constancy based in the diagonal-offset method and k-means is described in Section 3. In Section 4, a method for the detection and location of an elongated object is shown. Finally, in Section 5 some conclusions and future work are presented.

## 2 Colors selection

Color selection is a crucial issue as the quality of rod color segmentation procedure highly relies on a good distinction among the rod colors and the others elements that typically appear in the scene. These are mainly the probes, colored in blue (Pantone 3015) and green (Pantone 363) as it is shown in Figure 2. Other relevant colors are the black and white of the markers and the red (Pantone 484) of the point of the welding gun.

After several experiments we decided four different colors would be used for rod point localization and orientation computation. Minimizing the number of different colors used in the rod allow us to choose a more contrasted palette that reduces errors during the segmentation step. In addition, different lengths of the color markers are used in order to facilitate the location of the rod point.

Therefore, color selection involves solving a placement problem: given the most frequently used colors, estimate the coordinates of new colors that maximizes the distance among the resulting color set (given and new). For practical reasons the xyz color coordinate system has been chosen, because in this space a good approximation for measuring color differences is obtained using the Euclidean distance.

The solution must be subject to an important restriction: not every reproducible color in the xyz system can also be reproduced in a printing system. The paint industry standard that offers one of the richer chromatic varieties is a color palette called Pantone. This color card offers a range of about 1,000 different colors. Thus our solution must be one color in this set. Figure 1 (left) represents the whole color set in the xy plane. It is remarkable that the entire gamut is bounded by a triangle. The figure also includes the position of the five given colors.

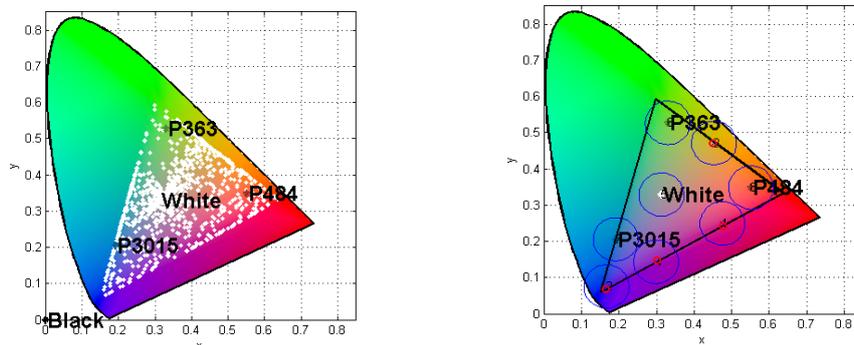


Fig. 1. Pantone full color set and position of the most frequent colors in the scene. Selected colors.

The problem can be stated more precisely as follows. Find the position of the center of a set of four circles that minimizes the area of the intersection among these circles and the circles defined by the position of the five fixed colors, subject to the center locations must establish a valid Pantone color.

The expression that relates relative position of the circles centers with the area enclosed by their intersection is a highly non-linear formula. Therefore, a genetic optimization algorithm has been selected to efficiently find the optimal placement. An important parameter is the diameter of the circles. Its length represents the distance from which we consider the colors are clearly distinguishable. In one hand, a low value can produce a solution with very similar colors, but on the other hand, a high value can turn the problem to unfeasible. Thus the optimization has been tested with a range of values for the radius R and we have discovered that a good compromise is obtained with R=0.06.

The center location restriction is imposed in two steps. In a first stage, circles centers are forced to lie inside the triangle that bounds the Pantone color set. In the second step, the nearest neighbor algorithm is used to find the Pantone color set nearest to the solution. If the distances were not small enough, the optimization algorithm with different parameters would be run again. However, as it is expected, solutions tend to be at the triangle borders, where the color availability is maximum. We have preferred to impose the restriction on the points lying inside the triangle rather than to allow only coordinates of Pantone colors during the optimization process as the first option can be more easily adapted to add new restrictions to the color set based on their visual appearance. The nonlinear constrains to restrict the solution to the points  $P$  inside a triangle defined by points  $A_1$ ,  $A_2$  and  $A_3$ , if it is based on sub-triangle orientations, can be written as follows. Let  $O_{123}$ ,  $O_{12P}$ ,  $O_{23P}$  and  $O_{31P}$  be the determinant of the matrix formed by stacking the coordinate differences for different pairs of points. More precisely

$$O_{123} = \det \begin{vmatrix} A_1 - A_3 \\ A_2 - A_3 \end{vmatrix},$$

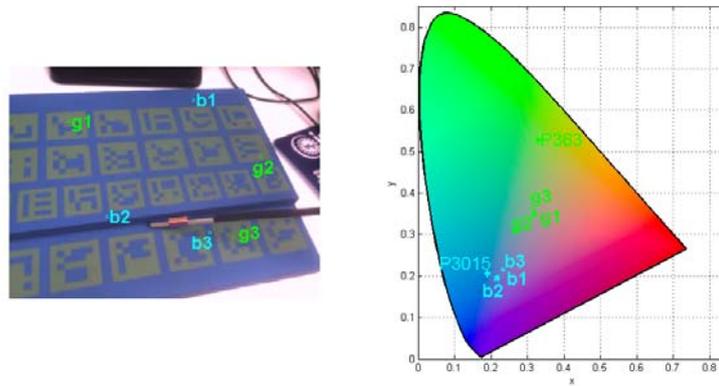
$$O_{12P} = \det \begin{vmatrix} A_1 - P \\ A_2 - P \end{vmatrix}, O_{23P} = \det \begin{vmatrix} A_2 - P \\ A_3 - P \end{vmatrix} \text{ and } O_{31P} = \det \begin{vmatrix} A_3 - P \\ A_1 - P \end{vmatrix}.$$

where  $P$  is inside the triangle defined by  $A_1A_2A_3$  if  $sign(O_{123}) = sign(O_{12P}) = sign(O_{23P}) = sign(O_{31P})$ , i.e., the sign of these determinants are the same.

Figure 1 (right) depicts the solution found by the genetic algorithm after running for a few hundred generations. The red symbol 'o' indicates the precise location of a valid Pantone color.

### 3 Color constancy

A video shot captured by the camera is shown in Figure 2. Several pixels have been annotated to show the deviation of the blue (Pantone 3015) and green (Pantone 363) colors under real illumination conditions. Blue pixels are relatively close to their ideal value, but green pixels have been biased and now they are closer to the blue values. Depending on the viewing angle and non-uniformity of the illuminant, this bias can be more or less high.



**Fig. 2.** Video shot captured by the system. Chromaticity values of several green (g1, g2 and g3) and blue (b1, b2 and b3) points are shown in a CIE chromaticity diagram in the right.

Most color constancy algorithms use the diagonal transform or von Kries Model (Hordley, 2006) to map the image gamut into a canonical gamut when certain conditions are met. More precisely, if only a single spatially uniform illuminant is considered, and objects are flat, coplanar, and Lambertian, that is, their reflectances are diffuse and independent from the angle of view. Unfortunately these conditions are not fulfilled in our experiments and, consequently, the diagonal transform cannot successfully map the image gamut into the canonical gamut. Thus, we have selected another model, the diagonal-offset model (Finlayson, Hordley, & Xu, 2005), which considers six parameters to map the image gamut. Let  $[R^c G^c B^c]$  be a color of the canonical gamut and  $[R^o G^o B^o]$  an observed color of the image gamut, then the diagonal-offset model maps this color using the next equation:

$$\begin{bmatrix} R^c \\ G^c \\ B^c \end{bmatrix} = \begin{bmatrix} d_1 & 0 & 0 \\ 0 & d_2 & 0 \\ 0 & 0 & d_3 \end{bmatrix} \begin{bmatrix} R^o \\ G^o \\ B^o \end{bmatrix} + \begin{bmatrix} o_1 \\ o_2 \\ o_3 \end{bmatrix} \quad (1)$$

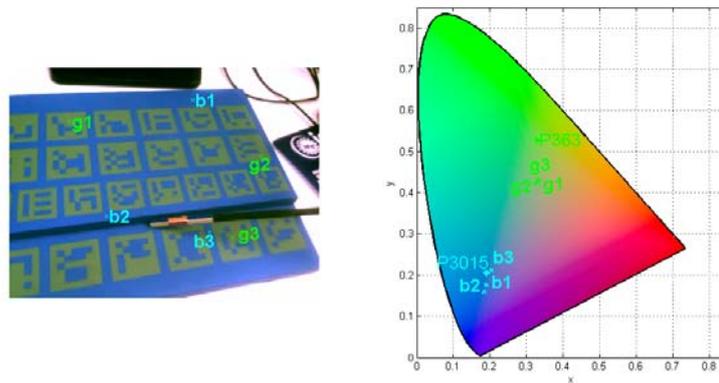
These six parameters can be obtained using convex programming but we have devised another method to reduce computational requirements and to improve the solution.

From our setup we know that most of the image is occupied by the blue and green working piece, plus the welding torch whose colors were selected in Section 2. Thus, the gamut of a typical frame in our application contains just a few colors, and some of them are known beforehand. Then, if we cluster the image gamut in several groups using k-means, for example with  $k=10$ , these clusters will likely correspond to the known colors and, moreover, two of the largest clusters will correspond to P3015 and P363 because the working piece occupies most of the frame.

We can select the centroids of the largest groups and substitute them into Equation 1, altogether with the RGB values of the green and blue Pantone colors, to get several six equations systems that allow directly computing the six parameters. For example, given the RGB values of P363 [76,140,43], P3015 [0,98,155], and the centroids of two clusters  $i [R^i, G^i, B^i]$  and  $j [R^j, G^j, B^j]$ , we can easily compute  $[d_1, d_2, d_3]$  and  $[o_1, o_2, o_3]$  by solving the next system:

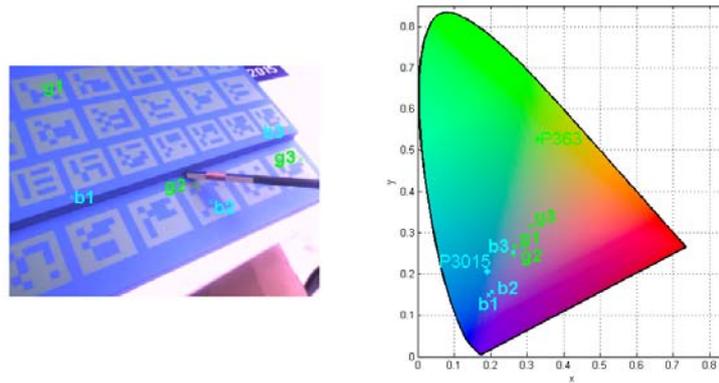
$$\begin{aligned} 76 &= d_1 * R^i + o_1 \\ 140 &= d_2 * G^i + o_2 \\ 43 &= d_3 * B^i + o_3 \\ 0 &= d_1 * R^j + o_1 \\ 98 &= d_2 * G^j + o_2 \\ 155 &= d_3 * B^j + o_3 \end{aligned}$$

Not every solution found by this way is a feasible illuminant, for example solutions with negative scaling factors ( $d < 0$ ) are unreal, thus some of them can be automatically discarded. On the other hand, some of the remaining solutions correspond to wrong illuminants, for example illuminants that transform real red values into blue values, and so on. These solutions can be discarded by computing the distance between the observed values and their expected canonical values, and removing those that are farther. And finally, as it is recommended in (Finlayson, Hordley, & Xu, 2005), we should look up among the remaining feasible solutions for the solution with a lowest offset vector.



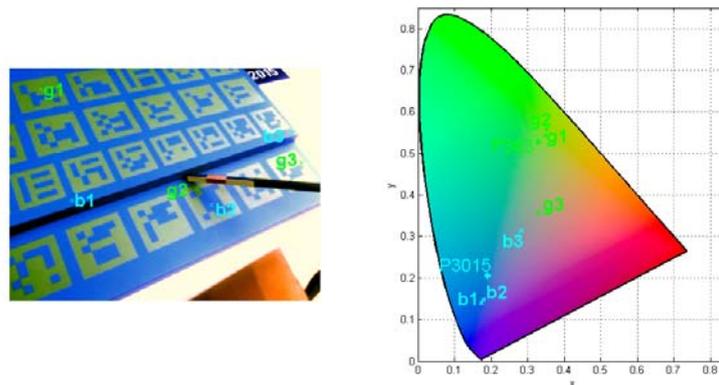
**Fig. 3.** Result of color correction algorithm on image of Fig. 1. Blue points are slightly closer to Pantone 3015, and green points are much closer to Pantone 363.

In Figure 3 it is shown the results obtained with this method in the image of Figure 2. Blue points remain close to Pantone 3015 and green points have been corrected to a much closer position to Pantone 363.



**Fig. 4.** A video shot under a bluish illuminant. Green and blue points are closer to Pantone 3015, and even one of the green points ( $g_2$ ) is closer than one of the blue points ( $b_3$ ).

A worse situation is shown in Figure 4. The scene was shot using a non-uniform bright bluish illuminant and most colors had been biased to the blue area. There are green points like  $g_2$  that are closer to Pantone 3015 than other blue points like  $b_3$ , rendering markers detection more difficult. Nevertheless, our color constancy method obtains a good result as it is shown in Figure 5. Green points in the left part of the image are now very close to Pantone 363, and the rest of green points are closer than before.



**Fig. 5.** After correcting image in Fig. 3, the green points are closer to P363 and the blue points have not been affected much.

Finally, to assess the computational requirements of our method, we have implemented it using the OpenCV library to measure the execution time. A full augmented reality application is typically multi-threaded, with specific threads for frame acquisition, processing, visualization, etc., thus we have developed a single-threaded version that can be easily incorporated into the full application. We have measured execution times between 5 and 7 ms in an Intel Xeon X5690 using RGB color frames of size 640x480, therefore real-time constraints can be met with our algorithm.

#### 4 Rod segments detection

Once the image has been corrected, the rod segments can be more easily detected. A different color was assigned to each segment as it was described in Section 2. To select which pixels belong to a specific color we use the distance metric defined by CIE (International Color Consortium), the CIEDE2000 color-difference formula (Sharma, Wu, & Dalal, 2005). For each color a mask is computed where pixels whose distance is below a threshold are selected. These pixels are grouped in blobs that correspond to the rod segments. They are somehow irregular and do not exactly represent a rectangular area as expected because, on the one hand, their size is too small for the spatial resolution of the camera and, on the other hand, the edges of the rod are not perfectly straight. Nevertheless, we can fit these blobs to rectangular areas and adjust the gaps by considering some geometrical constraints.

First, we fit each blob to an ellipse using the method presented in (Fitzgibbon, Pilu, & Fisher, 1999). The ellipsis of blobs corresponding to rod segments should be approximately aligned and very close among them. If there is a small gap between two contiguous ellipsis, they are both elongated to obtain two touching ellipsis. Then, the rotated rectangles that contain these ellipses can be computed to obtain the rod segments.



**Fig. 6.** Result obtained searching for the rod segments. Left, color corrected video shot. Center, ellipsis detected. Right, corners of rod segments.

In Figure 6 it is shown the result of searching for Pantone 395 (lime) and Pantone 484 (red) in a video frame. Color corrected video shot is shown in the left part of the figure. In the middle part, the ellipsis corresponding to the three rod segments are drawn. These ellipsis are not perfectly aligned and do not entirely occupy the segment area, thus they are elongated to obtain a better approximation. In the right part of Figure 6, the corners of the detected rod segments are marked.

Although there are some small errors in the location of the corners, they can be cancelled when the homography is computed for all these points.

## 5 Conclusions

In this paper we have presented a method for improving a tracking algorithm in an augmented reality application. Our method addresses several issues to this particular application, like marker-less tracking and color constancy with low quality cameras, or precise tracking with real-time constraints. We have presented a method for selecting colors of the objects in the scene and a new algorithm for color constancy based on the diagonal-offset method and k-means. Our method significantly improves the color information in the scene, making feasible a color based tracking algorithm.

In a future work we pretend to improve our color constancy method using other illumination models. We also plan to enhance tracking precision by using super-resolution techniques.

## Acknowledgements

This work has been partially supported by the Ministry of Education of Spain (TIN2013-42253P) and the Junta de Andalucía of Spain (TIC-1692). We also gratefully thank Seabery for providing the camera and equipment.

## References

- Azuma, R., Baillot, Y., Behringer, R., Feiner, S., Julier, S., & MacIntyre, B. (2001). Recent advances in augmented reality. *Computer Graphics and Applications*, 21(6), 34-47.
- Bower, M., Howe, C., McCredie, N., Robinson, A., & Grover, D. (2013). Augmented reality in Education - Cases, places, and potentials. *Educational Media (ICEM)* (págs. 1-11). Singapore: IEEE.
- Buchsbaum, G. (1980). A spatial processor model for object colour perception. *Journal of the Franklin Institute*, 1-26.
- Finlayson, G., Hordley, S., & Xu, R. (2005). Convex Programming Colour Constancy with a Diagonal-Offset Model. *International Conference on Image Processing*, (págs. 948-951).
- Fitzgibbon, A., Pilu, M., & Fisher, R. B. (1999). Direct Least Square Fitting of Ellipses. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(5), 476-480.
- Hordley, S. (2006). Scene Illumination Estimation: Past, Present and Future. *Color Research & Application*(31), 303-314.
- Sharma, G., Wu, W., & Dalal, E. (2005). The CIEDE2000 color-difference formula: Implementation notes, supplementary test data, and mathematical observations. *Color Research & Application*(30), 21-30.