A comparative study of grayscale conversion techniques applied to SIFT descriptors

Samuel Macêdo
Centro de Informática
UFPE
Email: svmm@cin.ufpe.br

Givânio Melo
Centro de Informática
UFPE
Email: gjm@cin.ufpe.br

Judith Kelner
Centro de Informática
UFPE
Email: jk@cin.ufpe.br

***Abstract*—In computer vision, gradient-based tracking is usually performed from monochromatic inputs. However, a few research studies consider the influence of the chosen color-to-grayscale conversion technique. This paper evaluates the impact of these conversion algorithms on tracking and homography calculation results, both being fundamental steps of augmented reality applications. Eighteen color-to-greyscale algorithms were investigated. These observations allowed the authors to conclude that the methods can cause significant discrepancies in the overall performance. As a related finding, experiments also showed that pure color channels (R, G, B) yielded more stability and precision when compared to other approaches.**

I. Introduction

Tracking algorithms based on descriptors often use images in grayscale to detect and extract features [11][13][2]. One of the reasons for using grayscale images instead of full color images is to simplify the three-dimensional color space (R, G and B) to a single dimensional representation, i.e., monochromatic. This fact is important because it reduces the computational cost and simplifies the algorithms. However, according to [10], the decision as to which color- to- grayscale mechanism should be used is still little explored. Studies tend to believe that, due to the robustness of the descriptors, the chosen grayscale conversion technique has little influence on the final result.

Several different methods are used in computer vision to perform color-to-grayscale conversion. The most common are techniques based on weighted averages of the red, green and blue channels, e.g., Intensity and Luminance [7]. Moreover, there are methods that adopt alternative strategies to generate a more accurate representation with respect to brightness, such as Luma and Lightness [9], although at first none of these techniques has been developed specifically for tracking and pattern recognition.

In [10], a case study is presented to demonstrate that there are significant changes in tracking results for different color-to grayscale conversion algorithms . The influences of these methods were also presented using SIFT and SURF descriptors ([12] and [2], respectively), Local Binary Patterns (LBP)[15] and Geometric Blur[3]. Nevertheless, the mentioned studies did not examine the red, green and blue pure channels as options for color-to-grayscale conversion.

Since color-to-grayscale conversion results are a function of the three pure channels (R, G and B), converted images lose part of the information they contained because a pixel in the input image, formerly represented by three dimensions in the color space, is then represented by a single one. Furthermore, there is no guarantee that the various color-to-grayscale conversion techniques converge to the same grayscale intensity value, i.e. the input image changes depending on the grayscale conversion method. Having said that, this paper aims to replicate the experiments on [10] and perform further experiments using the R, G and B pure channels as grayscale conversion techniques themselves.

II. Methods

This section describes basic concepts involved in the experiments and results presented in this article. It encompasses the explanation of color system concepts, brightness correction, color-to-grayscale conversion, SIFT and matching.

A. Color systems

Color-to-grayscale conversion algorithms all generate a monochromatic representation as output of the original color image. There are many color space representations, each one designed for a specific purpose and with its own coordinate system, for example, CYM[7], CYMK[7], RGBE[6], RGBW[5], HSL[1], HSV[1] and HSI[1]. The most common way of representing a pixel in a color space is RGB[7].

HSL and HSV are other common representations of points in the color space. These color space representations are cylindrical in shape and aim to rearrange the geometry of the color space in an attempt to be more intuitive than the cube-shape representation employed in the RGB model. The representation of the color space in HSL, HSV and RGB versions can be seen in Figure 1

B. Gamma Correction

Gamma correction is a nonlinear operation used to control signal amplitude. In digital image processing, this operation is used to control brightness and reduce gradient variation in video or still image systems. Human vision, under common illumination conditions (neither pitch black nor blindingly bright) follows an approximate gamma function, with greater sensitivity to relative differences between darker tones than lighter ones. The gamma correction follows the power law[16] according to the equation 1.

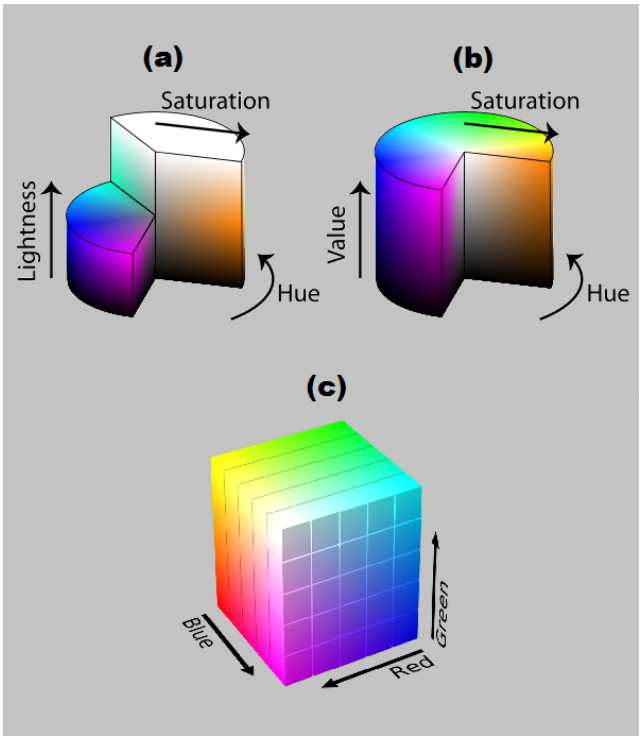

Fig. 1. (a) HSL (b) HSV (c) RGB

$$x' = \Gamma(x) = A * x^{\gamma} \quad (1)$$

where x is the pixel intensity in the image matrix representation, A is a scalar and γ is the correction parameter. The usual value for A and γ is 1 and $1/2.2$ respectively[16]. The figure 2 displays examples of different A and γ. We denote gamma corrected channels as R, G, and B. The gamma correction may also be applied to a function output - for example, the Luma algorithm corresponds to the Luminance algorithm using gamma corrected input (R', G' and B').

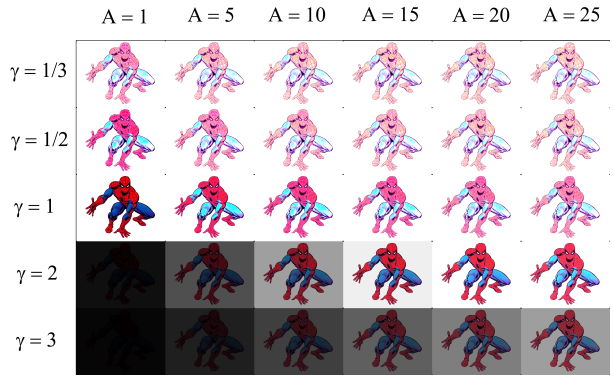

Fig. 2. Original figure at A = 1 and γ = 1

C. Color-to-Grayscale Conversion Algorithm

This section will briefly describe the color-to-grayscale methods studied during this research. The most popular approach to the problem of conversion is the Luminance algorithm, which approximates the image gradient to the human vision perception[4]. Every grayscale conversion technique is a function $\mathcal{G}$ that receives a colorful image as input $\mathbb{R}^{3mn}$ and outputs a monochromatic image $\mathbb{R}^{mn}$. Assuming all digital images used in this research are typically 8-bit per channel, the discrete pixel values $\updownarrow$ respect a limit of $L = 2^8$ possible levels of intensity, in other words, all values of the color input and the grayscale output are located between 0 and 255. In this scale, 0 represents black and 255 represents white. It is also assumed that R, G and B stand for a linear representation of the red, green and blue channels respectively. The color-to-grayscale conversion produces a pixel matrix with values between 0 and 255. The table I shows the conversion algorithm used in [10].

TABLE I
COLOR-TO-GRAYSCALE ALGORITHM

$\mathcal{G}_{Luminance}$	$0.21 * R + 0.71 * G + 0.07 * B$
$\mathcal{G}_{Intensity}$	$\frac{R+G+B}{3}$
$\mathcal{G}_{Value}$	$argMax(R, G, B)$
$\mathcal{G}_{Lightness}$	$\frac{1}{100}(116\{(Y) - 16)$
$\mathcal{G}_{Luster}$	$\frac{argMax(R,G,B)+argMin(R,G,B)}{2}$
$\mathcal{G}_{Luma}$	$0.21 * R' + 0.71 * G' + 0.07 * B'$
$\mathcal{G}_{Gleam}$	$\frac{R'+G'+B'}{3}$
$\mathcal{G}'_{Luminance}$	$\Gamma(G_{Luminance})$
$\mathcal{G}'_{Intensity}$	$\Gamma(G_{Intensity})$
$\mathcal{G}'_{Value}$	$\Gamma(G_{Value})$
$\mathcal{G}'_{Lightness}$	$\Gamma(G_{Lightness})$
$\mathcal{G}'_{Luster}$	$\Gamma(G_{Luster})$

In addition to replicating the experiments, this paper also extends the research adding six more grayscale conversion techniques: the pure channels (R, G and B) and the pure channels corrected by gamma function (R', G' and B'). The idea is to evaluate tracking behavior using a grayscale that is an actual output of the camera and not a function of the values given by this camera.

D. Scale Invariant Feature Transform

The image feature generation used in Lowe's method transforms an image into a large collection of feature vectors, each of which describes a point, named keypoint, in the image. These vectors are named as descriptors and they are invariant to translation, scaling, and rotation, partially invariant to illumination changes and robust to local geometric distortion. Firstly, the keypoints locations are defined as maxima and minima according to Difference of Gaussians (DoG) function applied in scale space to a series of smoothed and resampled images[12]. Low contrast candidate points and edge response points along an edge are discarded using interpolations of the samples and the Hessian matrix[12]. Dominant orientations are assigned to localized keypoints. These steps ensure that the keypoints are more stable for matching and recognition. SIFT descriptors are then obtained by considering pixels around a radius of the key location.

E. Feature matching

The SIFT descriptors are stored and indexed, and then they are matched with the SIFT descriptors in the other image. The best candidate match for each keypoint is found by identifying its nearest neighbor in the database of keypoints from training images. The nearest neighbors are defined as the keypoints with minimum Euclidean distance from the given descriptor vector. But in some cases, the second closest-match may be very near to the first. It may happen due to noise or some other reasons. In that case, the ratio of closest-distance to second-closest distance is taken. If it is greater than 0.8, the match is rejected. It eliminates around 90% of false matches while discarding only 5% correct matches[12]. The non rejected points are named a good match.

III. Experiment

The experiment performed in this paper aimed to study the influence of grayscale conversion in descriptor-based tracking behavior and its results. This work focused on the influence of different grayscales on SIFT descriptors [12].

For each image chosen as template, four points were selected as shown in Figure 3. These points were chosen because they have no detectable feature characteristics, i.e. they could only be estimated using homography calculation. Each tracked template path was compared to a correspondent ground truth, which specifies the real path of these four selected points along the video frames.

The performance of different color-to-grayscale conversion techniques was analyzed based on their ability to maintain object tracking throughout the video. In other words, for a particular grayscale conversion algorithm, the number of frames which established sufficient correlation between the template and the actual scene was verified. This allows for homography estimation and the identification of pre-established points in the figure 3. These were the points compared to the ground truth. Algorithms were ranked regarding their tracking stability using the equation 2 as a score function.

Fig. 3. Points on groundtruth

$$s = \frac{10}{v} \sum_{i=1}^{v} \frac{n_i}{N_i} \tag{2}$$

where v is the number of tested videos, n_i the number of frames in the video i where SIFT obtained sufficient features for homography calculation, and N_i the total number of frames in the video i. The experiment is summarized in the algorithm 1.

Algorithm 1 Experiment Steps

$T_{grayscale} = \mathcal{G}_i(Template)$
Feature extraction of $T_{grayscale}$
Generate SIFT descriptor using $T_{grayscale}$
for All video frames **do**
  $Q_{grayscale} = \mathcal{G}_i(Quadros)$
  Feature extraction of $Q_{grayscale}$
  Generate SIFT descriptor using $Q_{grayscale}$
  $M =$ Amount of feature matching
  if $M \geq 8$ **then**
    Pose estimation and corner points estimation
  end if
  Campare with the groundtruth
end for

IV. Results

The experiment was performed using five videos for each of the four templates, 20 samples in total. Each video had an average time of 10 seconds recorded at 30 fps. The videos

were named with the template initials followed by a number. The templates used can be seen in Figure 4.

Fig. 4. (a) Template 1: Book, (b) Template 2: Spider-Man, (c) Template 3: Green Lantern, (d) Template 4: Modified Green Lantern

To increase the experiment robustness, each video was filmed randomly according to the following categories:

- Format
  - .avi
  - .mov
- Compression
  - Raw
  - MPEG-4
- Scene
  - Moving
  - Static
- Camera
  - Microsoft - Lifecam 1393
  - Canon T4I
- Ilumination
  - Artificial (IRC 70%)
  - Natural (Sun)

The set of all videos and specifications can be seen in table II.

A. Good matching

To calculate the homography it is necessary to have at least four correspondence points[8]. The first part of the experiment is to verify whether all grayscales are capable of producing enough good matching to calculate the homography. The percentage of frames with enough good matching for templates 1, 2, 3 and 4 are in TABLE III, IV, V and VI , respectively.

As you can see in TABLE III, IV, V and VI, all color-to-grayscale algorithms are able to produce enough good matches in a majority of frames. But the results in the next section show that good matching does not mean good homography.

TABLE II
SAMPLES SPECIFICATIONS

Video	Format	Compression	Scene	Camera	Illumination
B.1	.avi	Raw	Mov	Lifecam	Art
B.2	.avi	Raw	Est	Lifecam	Art
B.3	.mov	MPEG-4	Est	Canon	Art
B.4	.mov	MPEG-4	Mov	Canon	Art
B.5	.mov	MPEG-4	Mov	Canon	Nat
SM.1	.avi	Raw	Mov	Lifecam	Art
SM.2	.avi	MPEG-4	Mov	Lifecam	Nat
SM.3	.avi	Raw	Est	Lifecam	Art
SM.4	.mov	MPEG-4	Est	Canon	Art
SM.5	.mov	MPEG-4	Est	Canon	Nat
GL.1	.avi	Raw	Mov	Lifecam	Art
GL.2	.avi	Raw	Est	Lifecam	Art
GL.3	.mov	MPEG-4	Est	Canon	Art
GL.4	.mov	MPEG-4	Est	Canon	Art
GL.5	.mov	MPEG-4	Mov	Canon	Nat
MGL.1	.avi	Raw	Est	Lifecam	Art
MGL.2	.avi	Raw	Est	Lifecam	Art
MGL.3	.mov	MPEG-4	Est	Canon	Art
MGL.4	.mov	MPEG-4	Est	Canon	Art
MGL.5	.mov	MPEG-4	Mov	Canon	Nat

TABLE III
TEMPLATE 1: % OF FRAMES WITH ENOUGH GOOD MATCHING

Algorithm	Videos				
	B.1	B.2	B.3	B.4	B.5
RED	100	100	100	100	100
GREEN	100	100	100	100	100
INTENSITY	100	100	100	100	100
GREEN'	100	100	100	100	100
LIGHTNESS	100	100	100	100	100
VALUE	100	100	100	100	100
INTENSITY'	100	100	100	100	100
VALUE'	100	100	100	100	100
GLEAM	100	100	100	100	100
LIGHTNESS'	100	100	100	100	100
LUSTER	100	100	100	100	100
RED'	100	100	100	100	100
LUSTER'	100	100	100	100	100
BLUE	100	100	100	100	100
BLUE'	100	100	100	100	100
LUMINANCE	100	100	100	100	99,70
LUMINANCE'	100	100	100	100	99,40
LUMA	100	100	100	100	98,79

B. Homography

The first template studied was Figure a. The default color-to-grayscale conversion algorithm in OpenCV[4] is Luminance. Considering OpenCV is widely used in computer vision applications, it was expected that this algorithm would have a satisfactory outcome. The results using the proposed score metric are related in table VII.

Note in Table VII that the pure red channel obtained the highest score (9.24) even in varying lighting, change of camera and/or motion blur. The pure green channel had similar results to the red in many cases with a score of 9.15. The algorithm

TABLE IV
TEMPLATE 2: % OF FRAMES WITH ENOUGH GOOD MATCHING

Algorithm	Videos				
	SM.1	SM.2	SM.3	SM.4	SM.5
GREEN	100	100	100	100	100
LUMINANCE	100	100	100	100	100
INTENSITY	100	100	100	100	100
GLEAM	100	100	100	100	100
LUSTER	100	100	100	100	100
RED'	100	100	100	100	100
BLUE'	100	100	100	100	100
BLUE	100	100	100	100	99,69
LUMA	100	100	100	100	99,69
VALUE	100	100	100	100	99,06
VALUE'	100	100	100	100	99,06
RED	100	100	100	100	98,13
GREEN'	100	99,75	100	100	100
LUSTER'	100	99,75	100	100	100
LIGHTNESS'	99,66	100	100	100	100
INTENSITY'	99,66	100	100	100	100
LIGHTNESS	97,24	100	100	100	100
LUMINANCE'	97,93	100	100	100	99,40

TABLE V
TEMPLATE 3: % OF FRAMES WITH ENOUGH GOOD MATCHING

Algorithm	Videos				
	GL.1	GL.2	GL.3	GL.4	GL.5
LIGHTNESS'	100	100	100	100	100
LUSTER	100	100	100	100	100
RED'	100	100	100	100	100
BLUE	100	100	100	100	100
GREEN'	100	100	100	100	99,69
VALUE	100	100	100	100	99,38
VALUE'	100	100	100	100	99,38
GREEN	100	100	100	100	99,38
LUMINANCE	100	100	100	100	99,38
INTENSITY	100	100	100	100	99,38
LUMA	100	100	100	100	98,45
LUMINANCE'	100	100	100	100	98,14
LUSTER'	100	100	100	100	98,14
INTENSITY'	100	100	100	100	97,52
GLEAM	100	100	100	100	97,21
RED	100	99,66	100	100	98,76
LIGHTNESS	99,33	100	100	100	99,69
BLUE'	100	99,33	99,66	100	99,38

TABLE VI
TEMPLATE 4: % OF FRAMES WITH ENOUGH GOOD MATCHING

Algorithm	Videos				
	MGL.1	MGL.2	MGL.3	MGL.4	MGL.5
LUMA	100	100	100	100	100
INTENSITY	100	100	100	100	100
VALUE	100	100	100	100	100
INTENSITY'	100	100	100	100	100
VALUE'	100	100	100	100	100
GLEAM	100	100	100	100	100
LUSTER	100	100	100	100	100
LUSTER'	100	100	100	100	100
BLUE'	100	100	100	100	100
LUMINANCE'	100	100	100	100	99,69
LIGHTNESS'	100	100	100	100	99,69
BLUE	100	100	100	99,38	100
RED	100	99,57	100	100	100
LIGHTNESS	100	99,57	100	100	100
RED'	100	99,14	100	100	100
LUMINANCE	99,66	100	100	100	100
GREEN'	100	99,14	100	100	99,69
GREEN	98,97	100	100	99,06	99,07

Luminance comes in third with a score of 8.69. As the Luminance is a weighted average of the three channels and the blue channel achieved a score of 0, the Luminance may have been influenced by the poor performance of the blue channel.

TABLE VII
TEMPLATE 1: % OF TRACKED FRAMES AND THE SCORE PROPOSED

Algorithm	Videos					Score
	B.1	B.2	B.3	B.4	B.5	
RED	92	100	100	70	100	9,24
GREEN	88	100	100	70	100	9,15
LUMINANCE	81	100	100	54	100	8,69
LUMA	75	100	100	46	6	6,55
INTENSITY	72	100	96	28	30	6,53
LUMINANCE'	66	100	100	39	4	6,18
GREEN'	71	100	100	26	9	6,12
LIGHTNESS	54	100	100	48	2	6,07
VALUE	26	100	64	5	91	5,70
INTENSITY'	36	100	95	30	0	5,20
VALUE'	26	100	40	4	88	5,16
GLEAM	41	100	85	18	0	4,88
LIGHTNESS'	18	100	31	13	0	3,26
LUSTER	11	99	0	0	1	2,23
RED'	1	100	0	0	0	2,03
LUSTER'	2	62	0	0	0	1,28
BLUE	0	0	0	0	0	0
BLUE'	0	0	0	0	0	0

Since the red channel had the best score and the blue one the worst, template 2 was chosen essentially because it is a red and blue image. Therefore, gradient difference is reduced, making it harder to extract image characteristics. Consequently, SIFT descriptors would perform badly. This template was used in order to exemplify a case where a single pure channel would not perform better than a function of all pure channels. The results using template 2 are in table VIII.

TABLE VIII
TEMPLATE 2: % OF TRACKED FRAMES AND THE SCORE PROPOSED

Algorithm	Videos					Score
	SM.1	SM.2	SM.3	SM.4	SM.5	
GREEN	100	70	100	100	100	9,39
RED	79	25	97	100	100	8,02
BLUE	95	29	88	95	100	8,13
LUMINANCE	3	61	100	11	100	5,50
INTENSITY	2	55	98	0	100	5,09
LUSTER	5	42	100	0	100	4,94
LUMA	0	23	0	0	96	2,38
LIGHTNESS	1	2	0	0	100	2,07
GLEAM	0	3	0	0	100	2,06
GREEN'	0	31	0	0	0	0,62
LUMINANCE'	0	9	0	0	0	0,18
BLUE'	0	0	1	0	0	0,03
VALUE	0	0	0	0	0	0
RED'	0	0	0	0	0	0
INTENSITY'	0	0	0	0	0	0
VALUE'	0	0	0	0	0	0
LIGHTNESS'	0	0	0	0	0	0
LUSTER'	0	0	0	0	0	0

The first noteworthy information in table VIII is the red channel performance, whose score decreased from 9.24 to 8.02, a drastic but expected result, as the template contained a lot of red. Hence, no considerable gradient differences existed. The green channel had the best performance in this test with a score of 9.39. This result might seem counter-intuitive at first, for the template used (figure 4b) was manipulated to have no green intensity; however, it is important to note that

all colors captured by CCD sensor models [14] (used in the majority of current digital cameras) are composed of R, G and B channels. That means that all pixels are composed by a combination of those three colors. Other relevant results are the blue channel performance (score 8.13) and the Luminance performance (score 5.50). The overall outcome suggests that, even when pure channels present good results, a function of these three channels (such as Luminance) will not necessarily present good results as well.

Template 3 was chosen in order to test the pure green channel, and thus, this template will naturally have little gradient difference in the green channel. For this test, it is expected that the pure green channel tracking performance would decrease as happened with the red channel in the previous test using template 2. Template 3 results are shown in table IX.

TABLE IX
TEMPLATE 3: % OF TRACKED FRAMES AND THE SCORE PROPOSED

Algorithm	Videos					Score
	GL.1	GL.2	GL.3	GL.4	GL.5	
RED	99	100	100	100	100	9,97
GREEN	91	100	26	100	88	8,10
INTENSITY	60	100	57	100	72	7,79
BLUE	92	100	13	100	66	7,44
LUSTER	83	100	33	100	49	7,28
LUMINANCE	19	100	4	100	15	4,77
VALUE	0	100	0	100	0	4,01
VALUE'	0	100	0	100	0	4,01
GLEAM	0	9	0	56	0	1,30
LUSTER'	0	44	0	3	0	0,94
INTENSITY'	0	31	0	4	0	0,69
BLUE'	0	21	0	2	0	0,47
RED'	0	5	0	11	0	0,32
LIGHTNESS	0	0	0	13	0	0,26
GREEN'	0	0	0	8	0	0,17
LUMA	0	0	0	7	0	0,15
LUMINANCE'	0	2	0	5	0	0,13
LIGHTNESS'	0	0	0	0	0	0

As shown in table IX, the red channel achieved the best performance in the group, its score being 9.97, similar to the test with template 1. As expected, the green channel had a similar performance to that of the red channel in the tests, its score decreasing from 9.39 to 8.10. Again, the traditional Luminance approach still reached a lower score than all pure channel approaches.

Based on the tests using templates 2 and 3, it was possible to notice that the predominance of a single color in a template may be prejudicial to SIFT descriptors and to feature-based tracking. Furthermore, multiple channel mix functions achieved a visibly lower performance when compared to pure channel approaches.

As modern cameras usually adopt the CCD system, at least one of the three primary channels should identify gradient variations that allow for feature extraction and tracking. To test this, an evaluation was conducted using a template that essentially presented only one channel.

Template 4 was produced by editing template 3 - it kept virtually none of its original blue and red intensities and suffered a green intensity boost. After this modification, expected results were to undermine the performance of the green channel and other green dependent approaches (Luminance for example). Experiment results can be examined in Table X.

TABLE X
TEMPLATE 4: % OF TRACKED FRAMES AND THE SCORE PROPOSED

Algorithm	Videos					Score
	MGL.1	MGL.2	MGL.3	MGL.4	MGL.5	
RED	98	46	26	18	100	5,78
LUSTER	40	81	1	3	61	3,73
INTENSITY	5	95	0	1	73	3,49
BLUE	28	4	0	2	34	1,36
GREEN	0	0	0	0	67	1,34
LUMINANCE	0	0	0	0	23	0,45
RED'	0	3	0	0	0	0,06
BLUE'	0	0	0	0	0	0,01
VALUE	0	0	0	0	0	0
LIGHTNESS	0	0	0	0	0	0
LUMA	0	0	0	0	0	0
GLEAM	0	0	0	0	0	0
GREEN'	0	0	0	0	0	0
LUMINANCE'	0	0	0	0	0	0
INTENSITY'	0	0	0	0	0	0
VALUE'	0	0	0	0	0	0
LIGHTNESS'	0	0	0	0	0	0
LUSTER'	0	0	0	0	0	0

As expected, the results in Table X display the low score achieved by all approaches, including the pure channels, as template 4 had almost no gradient variation. Green channel scores plummeted from 8.10 to 1.34, a predictable result given the previous template manipulation, and the red channel had the best result with a 5.78 score. Luminance scored 0.45, still lower than all pure channel approaches, and was surpassed by Intensity (score 3.49) because the green channel's influence on this method is lower.

The next step of the research was to evaluate the precision of point estimation when compared to the ground truth. Figure 5 shows a path estimated using SIFT descriptors compared to the ground truth. This analysis used the mean squared error (MSE) technique for each template in each video as a metric to evaluate tracking precision. Results are displayed in Table XI.

TABLE XI
TEMPLATES 1,2,3 AND 4 EQM

Algorithm	Templates			
	1	2	3	4
GREEN	1,33	1,23	1,18	8,47
RED	1,52	1,34	2,71	7,43
BLUE	-	1,95	8,86	8,13
INTENSITY	2,44	1,79	1,53	9,51
LUMINANCE	1,5	2,02	4,67	5,37
LUMA	1,61	-	-	-
GREEN'	3,23	-	-	-
LUMINANCE'	5,35	-	-	-
INTENSITY'	5,04	-	-	-
LIGHTNESS	6,57	-	-	-
GLEAM	8,65	-	31,67	-
VALUE'	53,2	-	18,79	2,00
RED'	46,25	-	-	-
VALUE	53,79	-	-	2,00
LIGHTNESS'	-	-	-	-
BLUE'	-	-	54,73	-
LUSTER	9,89E+06	1,18	2,22	9,84
LUSTER'	1,90E+07	-	-	-

As seen in Table XI, only the red, green, Luminance, Intensity and Luster are able to calculate homography in all

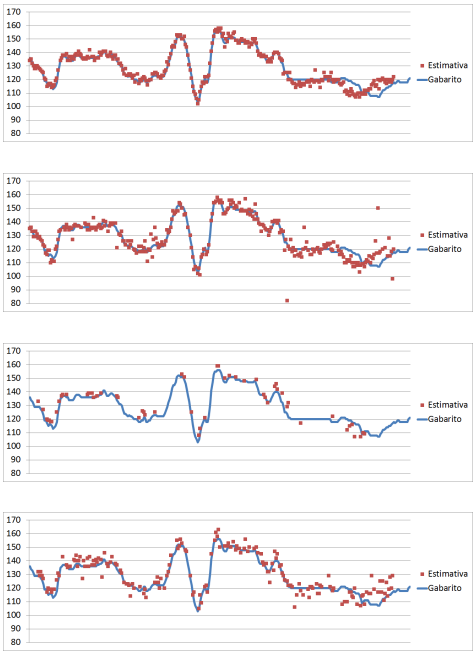

Fig. 5. Left superior point estimated path on video MGL.5 y coordinate (a)Pure red channel, (b) Pure green channel, (c) Luminance, (d) Intensity.

cases. Among these five algorithms, the pure channels red and green perform better than the others because they are those with the lowest mean squared error (MSE) and only obtained inferior results in the synthetic case (experiment with template 4).

V. Conclusion

The initial results show a significant variation on SIFT output and performance according to the grayscale method used to process the input frame. After comparing results, one can point out that pure channels (R, G and B) are better than other approaches, generating numerically consistent outputs, which proved to be very effective for tracking using SIFT descriptors.

The computational cost to perform this type of tracking is negligible compared to all the algorithms already implemented, considering the absence of any test, operation or adjustment beyond the direct assignment of the value of the channels. Among primary channels, the top performers were the red and green channel. The blue channel had an unsatisfactory performance when compared to the red and green channels.

VI. Future Works

The next step in this research is to evaluate the influence of light sources and camera sensor standards in the SIFT descriptors. Other subjects of interest are the influence of grayscale conversion approaches in other feature description techniques as well as the implementation of a hybrid and adaptive grayscale conversion, robust to color and lighting variation.

References

[1] M. K. Agoston. *Computer graphics and geometric modeling: implementation and algorithms*. Springer, London, 2005.

[2] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Surf: Speeded-up robust features. *Comput. Vis. Image Underst.*, 110(3):346–359, jun 2008.

[3] A. C. Berg, T. L. Berg, and J. Malik. Shape matching and object recognition using low distortion correspondences. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1 - Volume 01*, CVPR '05, pages 26–33, Washington, DC, USA, 2005. IEEE Computer Society.

[4] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

[5] Eastman Kodak Company. New kodak image sensor technology redefinesdigital image capture, 2007.

[6] Sony Corporation. Realization of natural color reproduction in digital still cameras, closer to thenatural sight perception of the human eye, 2003.

[7] R. C. Gonzalez and R. E. Woods. *Digital Image Processing*. Prentice-Hall, Inc., 2006.

[8] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer*. 2nd ed. Cambridge University Press, ISBN: 0521540518, 2004.

[9] K. Jack. *Video Demystified: A Handbook for the Digital Engineer, 5th Edition*. Newnes, Newton, MA, USA, 5th edition, 2007.

[10] C. Kanan and G. W. Cottrell. Color-to-grayscale: Does the method matter in image recognition? *PLoS ONE*, 7(1):e29740, 01 2012.

[11] D. G. Lowe. Object recognition from local scale-invariant features. In *Proceedings of the International Conference on Computer Vision-Volume 2 - Volume 2*, ICCV '99, pages 1150–, Washington, DC, USA, 1999. IEEE Computer Society.

[12] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110, nov 2004.

[13] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(10):1615–1630, oct 2005.

[14] J. Nakamura. *Image Sensors and Signal Processing for Digital Still Cameras*. CRC Press, Inc., Boca Raton, FL, USA, 2005.

[15] T. Ojala, M. Pietikäinen, and D. Harwood. A comparative study of texture measures with classification based on featured distributions. *Pattern Recognition*, 29(1):51–59, jan 1996.

[16] C. Poynton. *Digital Video and HDTV Algorithms and Interfaces*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1 edition, 2003.