

# Seguimiento Automático para RGB y Detección de Objetos en Color

Henry Cruz, Juan Meneses, Martina Eckert, José Martínez

Centro de Investigación en Tecnologías de Software y Sistemas Multimedia para la Sostenibilidad (CITSEM)  
Universidad Politécnica de Madrid  
Madrid, España

**Resumen** – Este artículo muestra un algoritmo de seguimiento (tracking) para RGB (red, green, blue) con un enfoque aplicativo para sistemas de video-vigilancia. Se ha desarrollado los procedimientos de segmentación, detección y finalmente el seguimiento así como la integración de información virtual en un ambiente real. Diversas pruebas experimentales se han realizado basadas en la propuesta de estimaciones de detección a través de comparación de longitud de onda del color y la distancia, así como considerando criterios utilizados en la literatura de precisión, rendimiento y valoración global de la correcta detección. Finalmente se ha podido comprobar que existe un rendimiento de detección a través de longitudes de onda del 83% así como una estimación de la métrica de precisión de la detección de 0.882, lo que permite demostrar una eficiente detección en el seguimiento a través algoritmo propuesto.

**Keywords**– segmentación, detección, seguimiento, RGB, precisión.

**Abstract** – This paper presents a RGB (red, green, and blue) algorithm in surveillance systems. Segmentation, detection, tracking and integration of virtual information has been implemented in the real environment. Experiments of comparing between color wave length and distance have been made. In addition, the criteria of recall, accuracy detection, F-measure are used to evaluate the accuracy of detections. Finally, we have obtained a good performance through length wave detection of 83% and F-measure of 0.882, these results confirm an accurate detection in the tracking through our algorithm.

**Keywords**– segmentation, detection, tracking, RGB, accuracy.

## I. INTRODUCCIÓN

Muchos métodos se han planteado en la obtención de características de la imagen, que permita realizar una segmentación eficaz, una detección adecuada y por consiguiente un seguimiento eficiente [1]. El objetivo del seguimiento es asociar un objetivo a un determinado número de cuadros de vídeo consecutivos con el fin de determinar la identificación, los lazos y la ubicación de los objetos en una secuencia de vídeo. Esta técnica, más conocida como *tracking* es ampliamente usada en la actualidad en diversas aplicaciones de distinta temática (detección facial, identificación biométrica, vigilancia y reconocimiento, etc.). Son numerosos los algoritmos desarrollados en los últimos años para la mejora de esta herramienta, entre otros, Viola-Jones, HOG y SVM, Haar like features, etc [2] [3] [4].

El objetivo principal de este trabajo es mostrar una contribución algorítmica modular para detección de colores implementado en MATLAB, la misma que es capaz de discernir entre las tres componentes básicas de color Rojo, Verde y Azul (RGB) y que tiene un enfoque a la contribución de técnicas aplicativas en relación a la video vigilancia, por lo que aspectos como disponer de una alta tasa de verdaderos positivos es fundamental. Los contenidos del presente artículo son los siguientes en el capítulo II se establece la metodología utilizada, describiendo el esquema general de implementación, el reconocimiento del color, el procesado y segmentación, la detección y etiquetado. Dentro del apartado III se presentan pruebas experimentales para establecer el porcentaje de eficiencia de la detección a través de dos métodos; uno de establecimiento de longitud de onda y otro a través del uso de criterios de precisión (*pr*) y recall (*re*). En el apartado IV se establecerán conclusiones así como trabajos futuros.

## II. METODOLOGÍA

En el tratamiento digital de la imagen, el uso de descriptores basados en el color permite discriminar los objetos que son de interés en diferentes espacios de color existentes. Los espacios de color no son más que formas de entender y tratar las imágenes en función de la finalidad para la que estén pensadas [5]. Algunos de los más relevantes pueden ser:

- RGB: Basado en las intensidades relativas al ojo de los colores rojo, verde y azul.
- YCbCr: Permite el tratamiento por separado de la luminancia y de la crominancia. Aprovecha el hecho de que el ojo es mucho más sensible a los cambios de luminancia para tratar esa parte con mayor importancia que la crominancia la cual se difunde con mayor compresión.
- HSV: Es una transformación no lineal del RGB. El espacio colorimétrico queda definido por tres variables: tono, saturación y valor.

En este estudio se va a utilizar el espacio de color en RGB debido se ajusta a cumplir sobre todo el requisito de mostrar tasas muy bajas de tiempos de ejecución. Para entender mejor este modelo, se va a caracterizar una imagen como una matriz de  $H \times W$  píxeles. En una imagen RGB, cada uno de estos píxeles será el resultado de la suma aditiva del mismo píxel de otras tres sub-imágenes (sub-matrices) que representan cada una de las componentes. Es común encontrar estas componentes

representadas por su luminosidad, de tal forma que, a mayor brillo de la imagen, mayor será la contribución de esa componente en ese punto en particular. En la Fig. 1 se muestra un ejemplo de cómo se produce la descomposición de RGB en escala de grises.

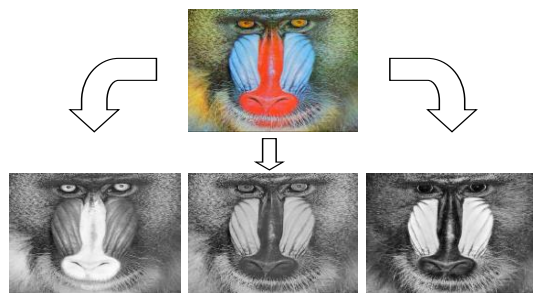


Fig. 1. Descomposición de una imagen RGB en sus tres componentes: rojo (izquierda), verde (centro) y azul (derecha)

**A. Esquema General de Implementación**

El esquema algorítmico que hemos planteado para la construcción del detector RGB se muestra en la Fig. 2.

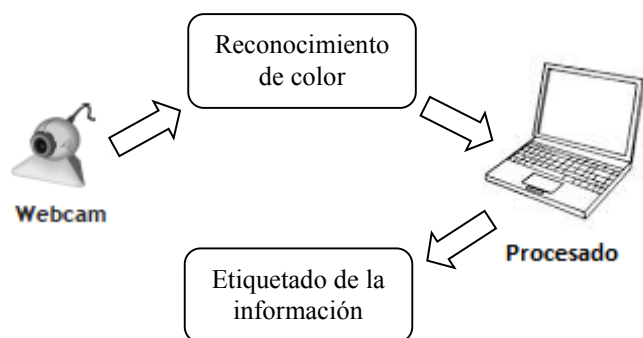


Fig. 2. Esquema del algoritmo empleado para detección RGB

En la Fig. 3 se muestra el pseudocódigo del funcionamiento del algoritmo implementado.

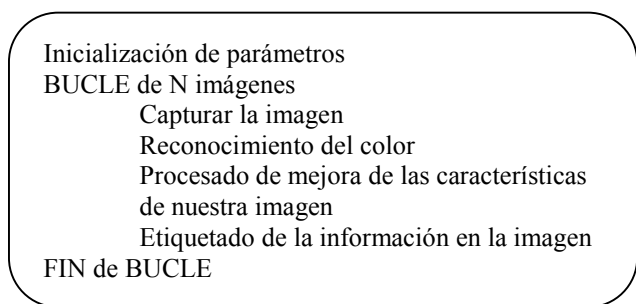


Fig. 3. Pseudocódigo del algoritmo empleado para detección RGB

El algoritmo comienza, con la configuración del elemento captador de imágenes (webcam), que es el medio de obtención de la información. Los parámetros a configurar en este apartado son la resolución con la que se nos mostrarán las imágenes capturadas, así como el modelo de color que vamos a

emplear y el intervalo en *frames* entre una captura y la siguiente. Para las pruebas de implementación se ha decidido tomar una resolución de 640x480 pixeles y con un intervalo entre capturas de 5 *frames*, esto por razones de rendimiento.

**B. Reconocimiento de Color**

Como se ha visto en párrafos anteriores, el modelo RGB nos permite desglosar una imagen en componentes de color, siendo muy habitual representar la luminosidad de las mismas. Para esta implementación nos hemos valido de este efecto para obtener la segmentación deseada a través de la operación aritmética de diferencia.

La adquisición de la información se realiza a través imagen en color obtenida por la webcam, a continuación, se obtiene la imagen de luminosidad de la imagen original y la imagen en escala de grises de la componente de color que queremos captar. Posteriormente realizamos una substracción de la información que contiene la imagen de luminosidad de la imagen original sobre la imagen de escala de grises de la componente a captar. Lo que se obtiene con esto, es una imagen en escala de grises en la cual quedan resaltados los objetos del color de la componente deseada. En la Fig. 4 se muestra un ejemplo para el caso de la componente rojo.

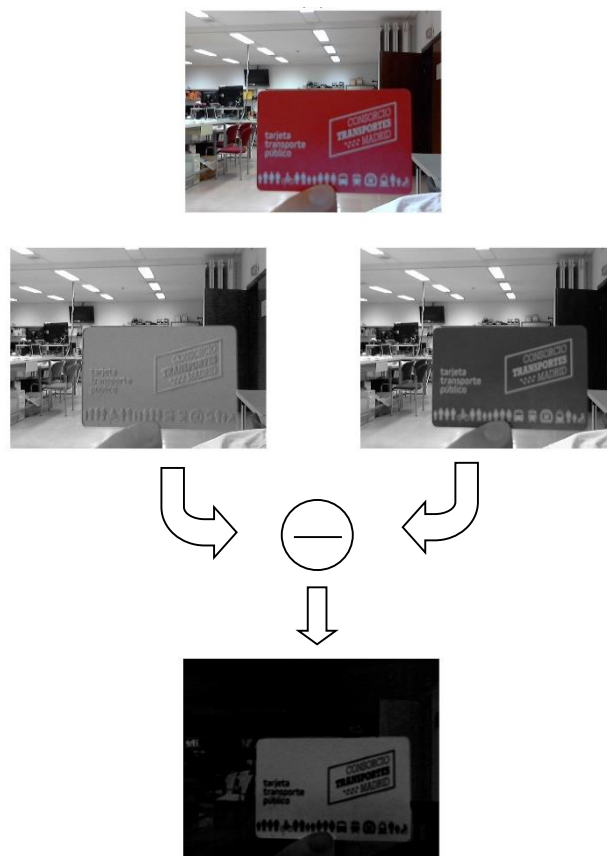


Fig. 4. Ejemplo reconocimiento color rojo. Imagen original (arriba), imagen luminancia banda de color (centro izquierda), imagen luminancia original (centro derecha) e imagen diferencia (abajo).

Hay que tener en cuenta que para calcular la señal de luminancia de la imagen original, es necesario utilizar la relación mostrada en (1).

$$Y = 0.299R + 0.587G + 0.114B \quad (1)$$

Como vemos la imagen de luminancia va a contener una mayor influencia por parte de la componente verde en relación a las otras. A la hora de hacer la substracción anteriormente explicada, el objetivo deseado es obtener la máxima diferencia (mayor brillo en la imagen de diferencia) para la componente a extraer. Esto nos permitirá una mejor identificación de los objetivos a seguir en el tracking.

A partir de (1) se puede apreciar que no todas las componentes van a proporcionar la misma imagen diferencia, en concreto en la componente verde existe una diferencia mucho más precaria, lo cual se habrá de considerar en la etapa de procesado. En la Fig. 5 se muestra diferencia de la componente verde sobre la imagen, en donde efectivamente se puede observar una menor cantidad de información proporcionada frente a las componentes azul y roja.

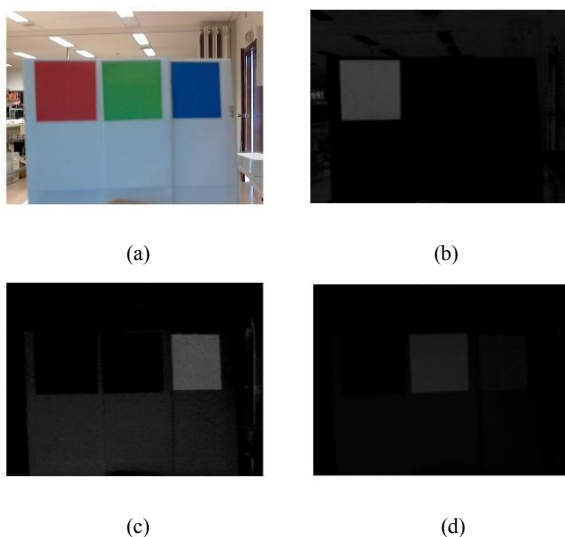


Fig. 5. Imágenes diferencia para las tres componentes (a) imagen original, (b) diferencia rojo, (c) diferencia verde, (d) diferencia azul.

### C. Procesado y Segmentación

El objetivo de esta etapa es mejorar la calidad de la imagen para la posterior identificación de los objetos dentro de la componente deseada y de su etiquetado. Básicamente son tres los procesos que se llevan a cabo:

1) *Filtrado*: Se realiza un filtrado que sustituye el valor de cada pixel de la imagen de diferencias por el valor medio compuesto por la matriz 4x4 de pixeles colindantes. Con esto conseguimos una mayor homogeneidad en la imagen, sobre todo en el fondo de la misma eliminando todos los posibles brillos producidos por objetos de no interés. En la Fig 6. Se puede observar el resultado del filtrado.



Fig. 6. Ejemplo de reconocimiento de color rojo después de filtrado

2) *Imagen binaria*: Básicamente se pasa de una imagen en escala de grises a una imagen en blanco y negro a través de la imposición de un umbral. Aquellos pixeles que superen el umbral se transforman en pixeles blancos y los que no en negros. Esta es la parte del procesado con la cual debemos conseguir que todas las componentes sean igualmente captadas, dando mayor umbral a los rojos y azules que serán más fáciles de detectar y menos a los verdes, ya que su imagen de diferencia tendrá menos brillo, una muestra se presenta en al Fig.7.

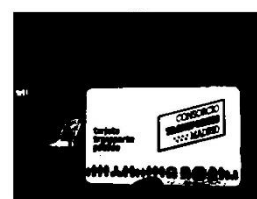


Fig. 7. Reconocimiento de color rojo pasada la imagen a binario.

3) *Eliminación de ruido por brillo y segmentación*: Si después del procesado anterior, aún quedasen pequeños restos de brillos blancos de fondo, éstos son clasificados para no tenerlos en cuenta en el posterior etiquetado. El resultado final de este procedimiento es la imagen segmentada del objetivo de interes. En la Fig. 8, se muestra el filtrado final.



Fig. 8. Ejemplo de reconocimiento de color rojo una vez eliminados los objetos de menos de 300 pixeles.

### D. Detección y Etiquetado.

La detección y etiquetado representan la última parte del proceso de tracking de colores en la que se integra la información al detector RGB. En este procedimiento podemos generar información virtual en un ambiente real por lo que existe amplia gama aplicativa. En este caso, se ha decidido recuadrar el contenido de la imagen coincidente con una de las tonalidades de cada una de las componentes de color, localizando el centro geométrico del mismo y mostrando por pantalla, tanto las coordenadas del objeto en referencia al conjunto de la imagen, como un pequeño texto explicativo de la componente que se está detectando, la Fig. 9 muestra el resultado final al integrar la información.

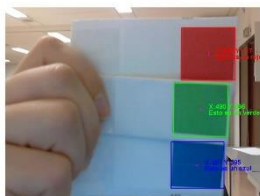


Figura 9. Ejemplo de etiquetado de colores y muestra de la integración de información virtual en ambiente real.

### III. EXPERIMENTACIÓN Y RESULTADOS

Para realizar las diversas pruebas de laboratorio se ha utilizado Matlab 2013 b instalado en un ordenador Intel Core i5 con 3.10 GHz y 7,89 Gb de RAM además de una webcam de 720pp HD. Para la evaluación de la detección se ha utilizado la base de datos 2014 Data Set [6]. La experimentación se ha llevado a cabo siguiendo dos procedimientos; primero a través de la estimación de detecciones por longitud de onda y la segunda a través de la estimación comparativa de precisión y recall.

#### A. Estimación de detecciones por longitud de onda

Esta valoración de las detecciones se realiza a través de la estimación comparativa propuesta del espacio de color y la longitud de onda respectiva frente a una distancia de evaluación. En la Tabla I se presenta las longitudes de onda que se han considerados para cada componente de color, estos valores son en general considerados como normativos pero pueden estar sujetos a variaciones por las condiciones ambientales en que se desarrollen los experimentos.

TABLA I. LONGITUDES DE ONDA PARA CADA COMPONENTE

Componente	Longitud de Onda (nm)
R	620 - 750
G	495 - 570
B	420 - 490

Para cada cierto valor de longitud de onda, se obtiene el color representativo y empíricamente se comprueba la detección y correcto etiquetado del mismo. Cada una de las capturas se realizó a varias distancias la Tabla II muestra los resultados obtenidos de las detecciones a una distancia de 20 cm de la cámara RGB para rojos. De igual forma en la Tabla III y Tabla IV se muestra los resultados obtenidos a través del mismo procedimiento para verdes y azules respectivamente.

TABLA II. RESULTADOS OBTENIDOS PARA ROJOS

Rojos		
Distancia (cm)	Detección correcta (nm)	Detección incorrecta (nm)
20	620 - 740	750
>20	620 - 740	750
<20	620 - 750	x
Verdaderos Positivos (%)	94	

TABLA III. RESULTADOS OBTENIDOS PARA VERDES

Verdes		
Distancia (cm)	Detección correcta (nm)	Detección incorrecta (nm)
20	495 - 535	540 - 570
>20	495 - 540	545 - 570
<20	495 - 550	555 - 570
Verdaderos Positivos (%)	62	
Falsos positivos	480 - 490 (nm)	

TABLA IV. RESULTADOS OBTENIDOS PARA AZULES

Azules		
Distancia (cm)	Detección correcta (nm)	Detección incorrecta (nm)
20	420 - 490	x
>20	435 - 490	420 - 430
<20	420 - 490	x
Verdaderos Positivos (%)	93	

De los resultados obtenidos se puede establecer que la evaluación es más estricta en los verdes con un 62% de verdes frente al 94% de rojos y 93 de azules%. El rendimiento global a través del porcentaje de aciertos es del 83%.

#### B. Estimación Comparativa de Precisión (*pr*) y Recall (*re*).

Para utilizar el criterio de *pr* y *re*, se emplea el modelo propuesto en [7]. Se ha realizado valoraciones de efectividad de detección a través de la toma de valores de verdaderos positivos ( $T_p$ ), verdaderos negativos ( $T_n$ ), falsos positivos ( $F_p$ ) y falsos negativos ( $F_n$ ) en 300 *frames* a través de pruebas comparativas en promedios parciales de 100 *frames* como se presenta en la Tabla V.

TABLA V. PROMEDIOS PARCIALES PARA 100 FRAMES

Test	Frames	$T_p$	$T_n$	$F_p$	$F_n$
1	100	90	87	10	13
2	100	87	86	13	14
3	100	89	90	11	10
Promedio total		88.67	87.68	11.31	12.37

Estos valores permitirán el cálculo de las tasas de *pr* y *re*, métricas mostradas a través de (2) y (3) respectivamente.

$$pr = t_p / (t_p + f_p) \quad (2)$$

$$re = t_p / (t_p + f_n) \quad (3)$$



A través de (2) y (3) se obtiene el valor de precisión de la detección  $F\text{-measure}$  ( $Fm$ ) [7] que consecuentemente permite establecer el rendimiento de la detección dentro del seguimiento como en (4).

$$Fm = 2 \left( \frac{pr \times re}{pr + rc} \right) \quad (4)$$

Con la disponibilidad de estas métricas se establece el rendimiento global de la detección del algoritmo propuesto a través del valor del coeficiente de correlación de Matthews ( $MCC$ ), usado con el mismo propósito en [8], también se ha considerado el cálculo de la tasa de verdaderos positivos y falsos positivos, que nos permitirá valorar a través de la curva de  $ROC$  la certeza de la clasificación y rendimiento [9]. Los resultados obtenidos se muestran en la Tabla VI.

TABLE VI. PROMEDIO DE EVALUACION DEL RENDIMIENTO DE DETECCIÓN SOBRE 300 FRAMES

pr	re	Tpr	Fpr	Fm	MCC
0.887	0.878	0.878	0.114	0.882	0.763

Los resultados obtenidos de las métricas  $pr$  y  $re$  son bastante aceptables y esto es debido a que al ser un sistema que tiene un modelo fijo de detección (RGB) y umbrales específicos, los resultados de la detección se garantizan. Esta premisa tiene soporte en base a los valores de  $Tpr$  y la baja estimación de  $Fpr$  así como en el valor presentado en  $Fm$ , la gráfica de rendimiento representada a través de la curva de  $ROC$  se muestra en la Fig. 10. Sin embargo  $MCC$  en general valora de forma más estricta pero representa un valor aceptable de evaluación del rendimiento global.

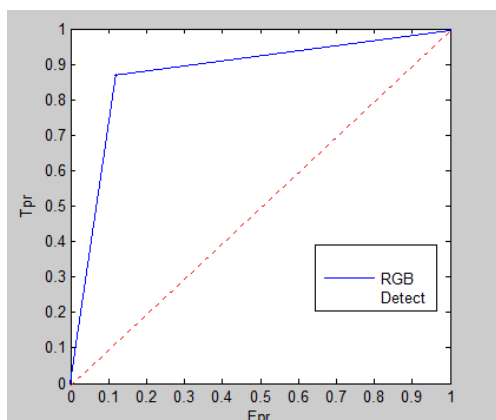


Fig. 10. Desempeño del rendimiento según  $ROC$

La curva de  $ROC$  se muestra un eficiente desempeño de la detección del algoritmo de seguimiento, considerando la métrica de valoración ideal es 1.

Es concluyente el establecer que la propuesta algorítmica de detección para RGB es funcional así como eficiente. Se ha obtenido un 83% de detecciones positivas en base a la propuesta de evaluación por longitud de onda. Por otro lado las métricas de evaluación de  $pe$  y  $re$  han mostrado valores de precisión y detecciones ciertas con un 0.887 y 0.878 respectivamente. También a través de los coeficientes de  $Fm$  (0.882) y  $MCC$  (0.763) se ha podido comprobar la eficiencia del rendimiento del seguimiento. Finalmente la valoración a través de los  $Tpr$  y  $Fpr$  ha permitido mostrar la curva de  $ROC$  con una respuesta que se acerca al valor ideal de 1.

Los trabajos futuros se plantean en las siguientes líneas:

- Implementación del detector de colores en un dispositivo de mano (cámaras, smartphones, tablets).
- Mejora de la eficiencia en la detección a través de técnicas de compensación de la iluminación.

En la primera línea, se enfrentarán problemas derivados de la implementación del algoritmo en cada uno de los diferentes sistemas operativos, así como escenas con movimientos significativos de *frame* a *frame* y eficiencia frente al consumo de baterías.

En la segunda línea, se enfrentarán problemas derivados de la captación de imágenes en entornos poco iluminados o no con las características más adecuadas.

#### ACKNOWLEDGMENT

Este trabajo fue patrocinado por el Plan Nacional Español para la Investigación Científica y la Innovación, proyecto número TEC2013-48453-C2-2-R.

Henry Cruz Carrillo agradece al Gobierno del Ecuador por la concesión de beca de estudios a través de la "Secretaría Nacional de Educación Superior Ciencia y Tecnología" (SENESCYT).

#### REFERENCIAS

- [1] E. Maggio and A. Cavallaro, "Video Tracking Theory and Practice", Edit. Wiley, 3rd Edition, 2011, pp. 3-120.
- [2] Z. Xin, Y. Yee-Hong, H. Zhiguang, W. Hui, G. Chao, "Object Class Detection: A Survey", Journal ACM Computing Surveys (CSUR), vol. 46, No 1, pp. 101-151, Oct. 2013.
- [3] Y. Li, S. Wang, Q. Tian, X. Ding, "A survey of recent advances in visual feature detection", Neurocomputing, vol. 149, pp. 736-751, Feb 2015.
- [4] J. Kwon and K. M. Lee, "Tracking of a non-rigid object via patch-based dynamic appearance modeling and adaptive basin hopping monte carlo sampling," in Proc. IEEE CVPR, Miami, FL, USA, 2009.
- [5] R. Hsu, M. Abdel-Mottaleb and A.K. Jain, "Face Detection in Color Images" In Proc. Of IEEE Workshop on Pattern Analysis and Machine Intelligence, 2002.
- [6] N. Goyette, P.-M. Jodoin, F. Porikli, J. Konrad, and P. Ishwar, "Changedetection.net: A new change detection benchmark dataset," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops, Jun. 2012, pp. 1-8.
- [7] Y. Wang, P. Jodoin, F. Porikli, J. Konrad, Y. Benezeth and P. Ishwar, "CDnet 2014: an expanded change detection benchmark dataset", 2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Columbus, pp. 393-400, June 2014.

- [8] P. St Charles, G. Bilodeau and R. Bergevin, "SuBSENSE: A Universal Change Detection Method With Local Adaptive Sensitivity", IEEE Transactions on Image Processing, vol. 24, no. 1, pp. 359-373, January 2015.
- [9] T. Fawcett, "An introduction to ROC analysis", Journal Pattern Recognition Letters, vol.27, iss. 8, pp. 861-874, Jun. 2006.