

Calibración de Cámara

Joel Gallegos, Maestría en Ciencias de la Computación, Universidad Católica San Pablo

□

Resumen: Este documento presenta paso a paso el proceso de calibración de una cámara, para ello se tiene un patrón de 20 anillos, con una distribución como matriz de 4x5. En este segundo avance se concluye el método de detección del patrón y se presenta un gran avance del rastreo del patrón

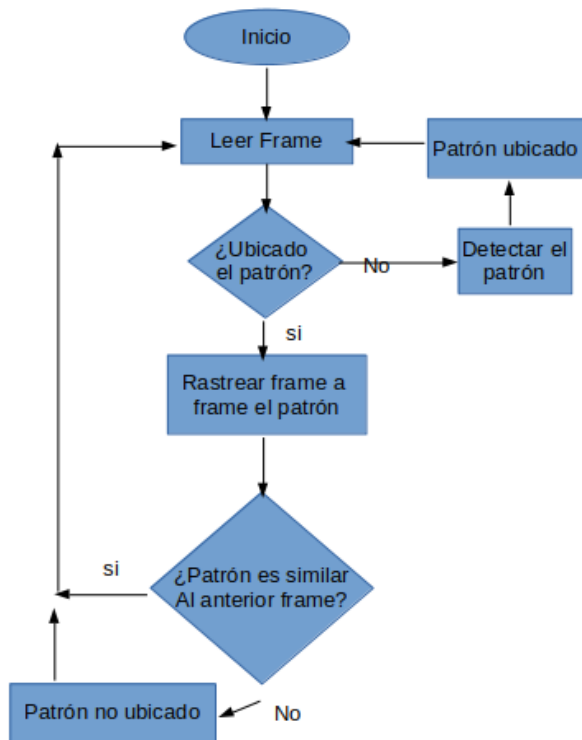
Palabras clave— Patrón de anillos, detección, rastreo.

I. INTRODUCCIÓN

La calibración de una cámara

II. DESARROLLO

Para la segunda parte del trabajo, se dividió el trabajo en dos partes, la primera es la detección del patrón de anillos y la segunda parte es el rastreo de los anillos, para ello miremos la siguiente imagen.



Detección del patrón

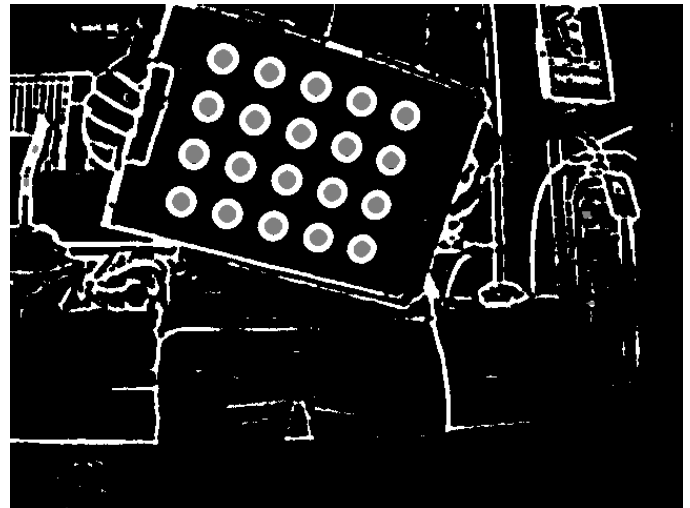
Primero se aplicó los siguientes métodos de procesamiento:

- Filtro blur.
- Binarización Adaptativa.
- Dilatación.

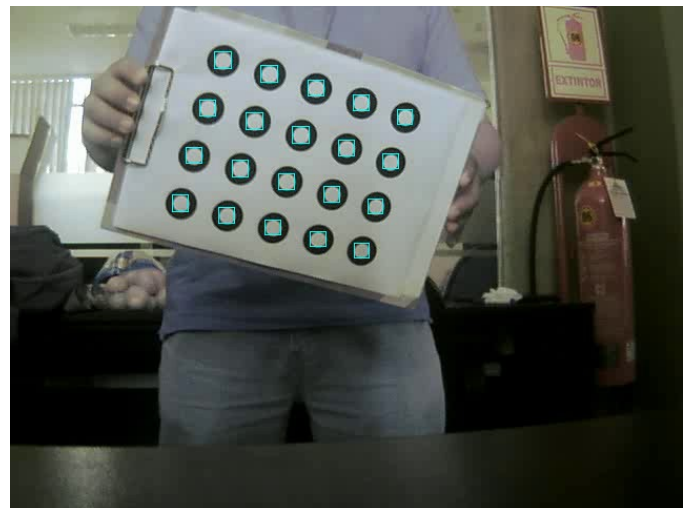
□

- Erosión.
- Detector de bordes con Canny.

Luego se detecto posibles elipses, los cuales podrian ser los anillos del patrón, gracias a este método, se hizo un relleno de polígono, considerando la forma del anillo, se obtiene un pixel central, y se rellena hasta encontrar su limite, para entenderlo mejor, se mira en la figura 2.

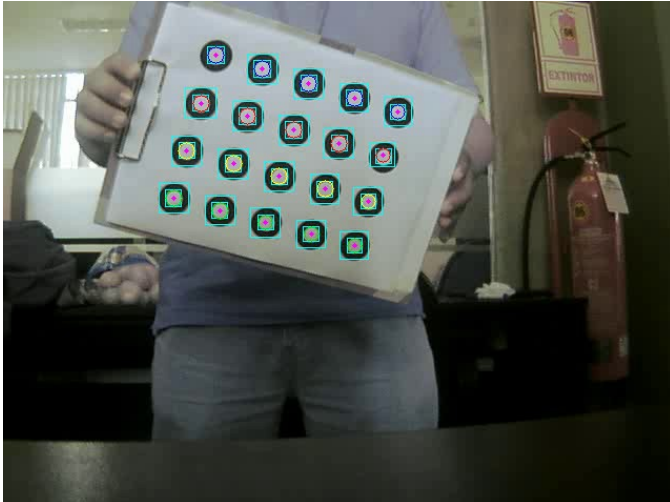


Gracias a que se obtuvo una imagen binarizada, se puede utilizar estos colores, el cual el centro en un comienzo era negro, pero como un relleno de polígono, se pinto el centro del anillo, considerando su color igual, hasta el limite blanco. Es así que se puede obtener un centro, un ancho y un alto del circulo interior del anillo.



Posteriormente se elimina algunas elipses encontradas, para

ello se considera la proximidad a otra elipse y el tamaño del mismo, y para concluir la detección de los anillos, se ubica el segundo circulo, que seria el circulo exterior del anillo, para ello se utiliza el mismo principio de colorear un poligono, es asi, que se obtiene un segundo centro ancho y alto, el cual sirve para corroborar y eliminar detecciones incorrectas.



Obteniendo finalmente los 20 anillos del patrón, considerar que para el siguiente paso se es necesario encontrar 20 anillos, caso contrario busca el siguiente frame.

Rastrear el patrón

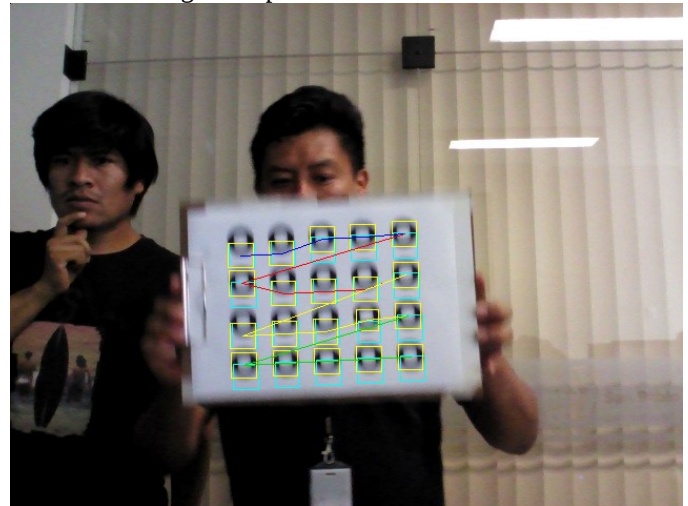
Para rastrear el patrón se utiliza el frame anterior y el actual, el frame anterior es para saber la ubicación del anillo, y el frame actual ver cuando se movió. Para ello se utiliza la imagen binarizada y se obtiene en un ROI los anillos, luego como la imagen es binaria, se utiliza los colores a nuestro favor, por ejemplo, si un anillo se mueve hacia la derecha se ubica en el ROI cuanto espacio deajo, como en el siguiente código:

```
int buscarEspacioDerecha(Mat roi)
{
    int espacioDerecha = 0;
    for(int x=roi.cols-1; x>=0; x--)
    {
        for(int y=0; y<roi.rows; y++)
        {
            if(roi.at<uchar>(y,x)==255)
            {
                espacioDerecha = (roi.cols-1)-x;
                x = 0;
                y = roi.cols;
            }
        }
    }
    return espacioDerecha;
}
```

Asi se actualiza las posiciones de nuestros anillos, como tambien sus nuevas dimensiones, si cambiara, miremos la siguiente imagen como ejemplo del desplazamiento del anillo,



Así se corrige la posición y tamaño, luego es necesario actualizar en cada frame la apariencia de cada anillo, pues nos siempre el anillo va a ser muy claramente visible, aquí es como se ve el siguiente problema.



Ante dicho problema tenemos dos posibles soluciones:

- La primera es hacer una pequeña búsqueda de cada anillo, lo cual se considera que esta cercano, tener en cuenta si aun así no se encuentran los anillos, se haría la segunda opción.
- La segunda es volver a ubicar el patrón.

Y así poder rastrear correctamente el patrón como en la siguiente imagen:

