

# Camera Calibration using Levenberg- Marquardt Algorithm

Student: Daniel Palomino



# Algoritmo



**Objetivo:**



Dado  $N$  imágenes conteniendo una grilla plana de calibración, estimar los parámetros de la cámara.

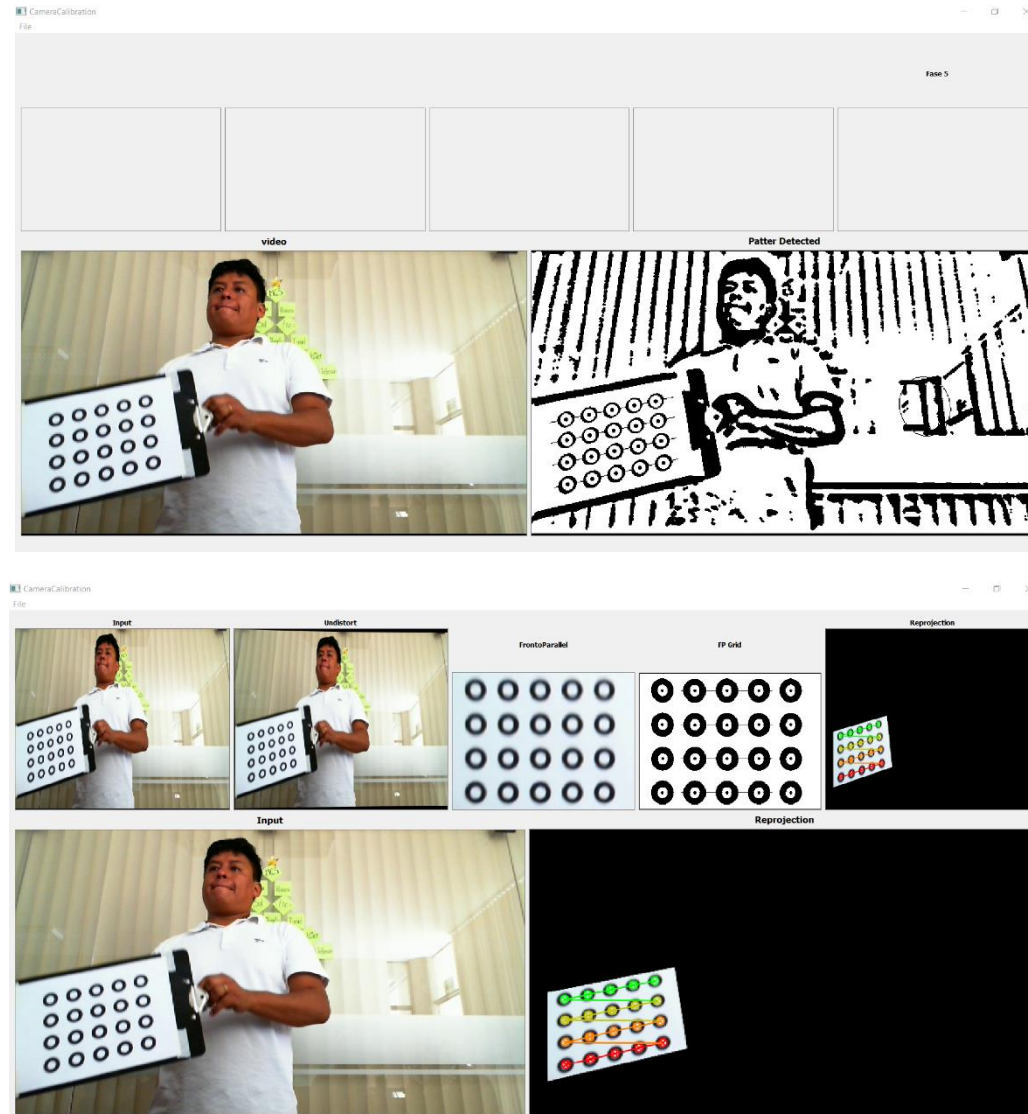
## Inicio:

- ▶ **Detectar los puntos de Control:** Detectar los puntos de control en el patrón de calibración (esquinas, círculos o centros de anillos).
- ▶ **Ajuste de Parámetros:** Usar los puntos de control detectados para estimar los parámetros de la cámara usando Levenberg-Marquardt.

### **Repetir hasta converger:**

- ▶ **Quitar la distorsión y la proyección:** Usar los parámetros de la cámara para quitar la distorsión y la proyección a las imágenes de entrada y obtener los padrones canónicos.
- ▶ **Ubicar los puntos de control:** Ubicar los puntos de control en el padrón de calibración canónico.
- ▶ **Reproyectar:** Proyectar los puntos de control usando los parámetros de la cámara estimados.
- ▶ **Ajusta de parámetros:** Usar los puntos de control proyectados para refinar los parámetros de la cámara usando Levenberg-Marquardt.

# Implementación



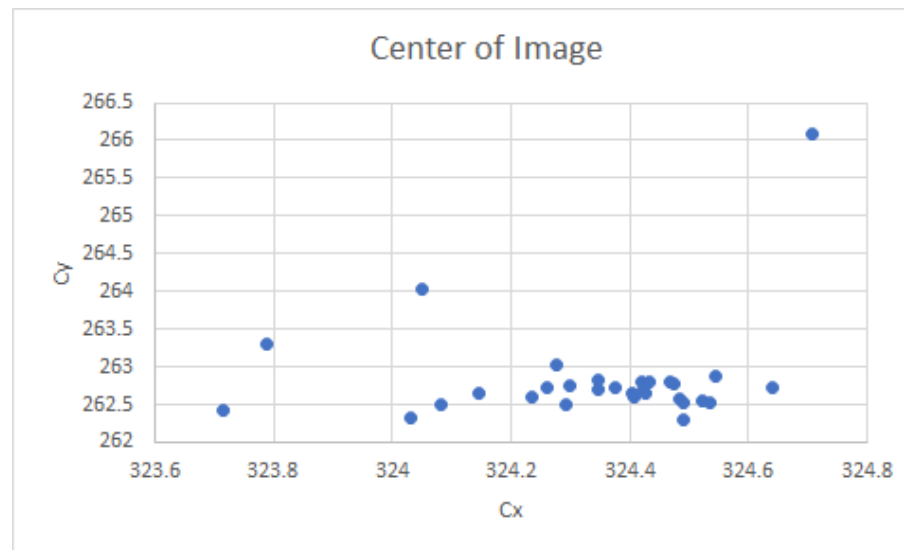
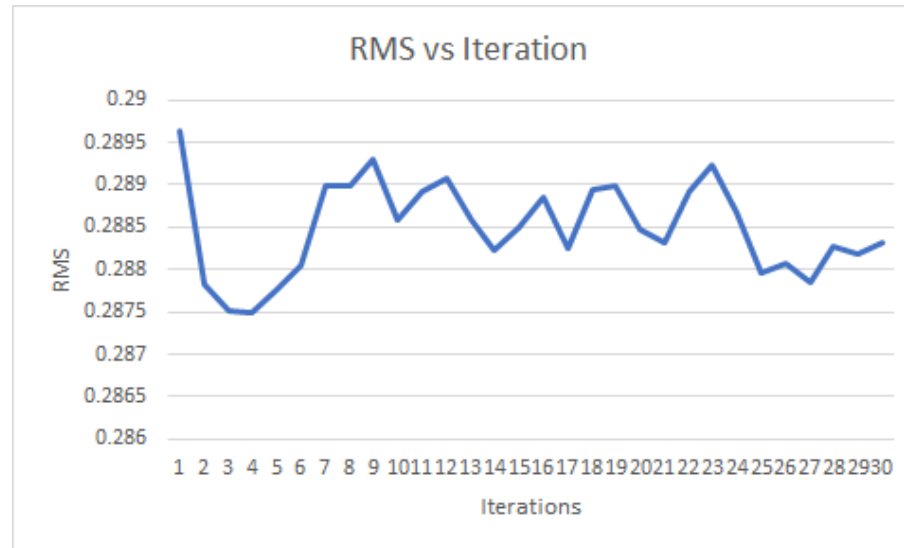
# Resultados

The background of the slide is composed of several overlapping, semi-transparent green geometric shapes. On the right side, there is a solid, vibrant green rectangular area. To its left, a series of diagonal, overlapping bands in various shades of green (from light lime to dark forest green) create a dynamic, layered effect. A thin, dark green line runs diagonally across the center, intersecting the various green shapes.



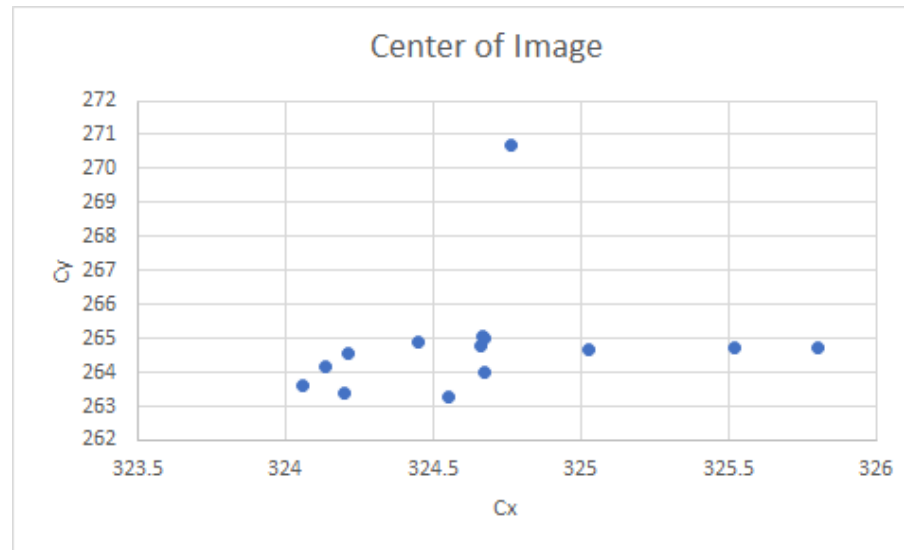
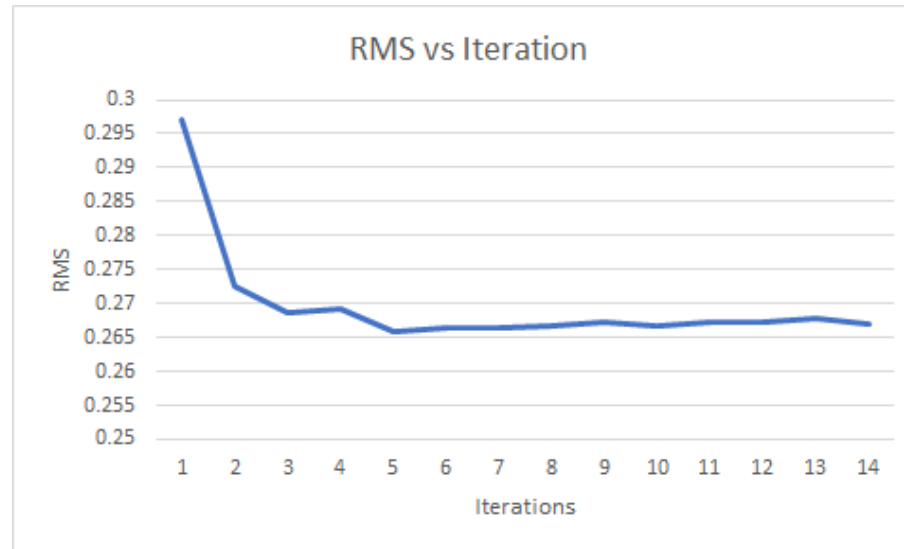
LiveCam  
Microsoft

# Algoritmo Iterativo Inicial





# Algoritmo Iterativo Final



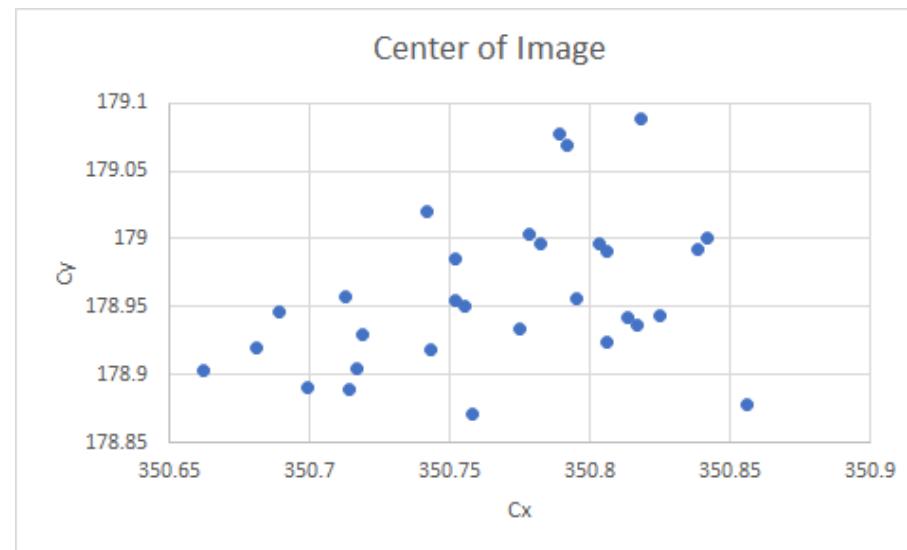
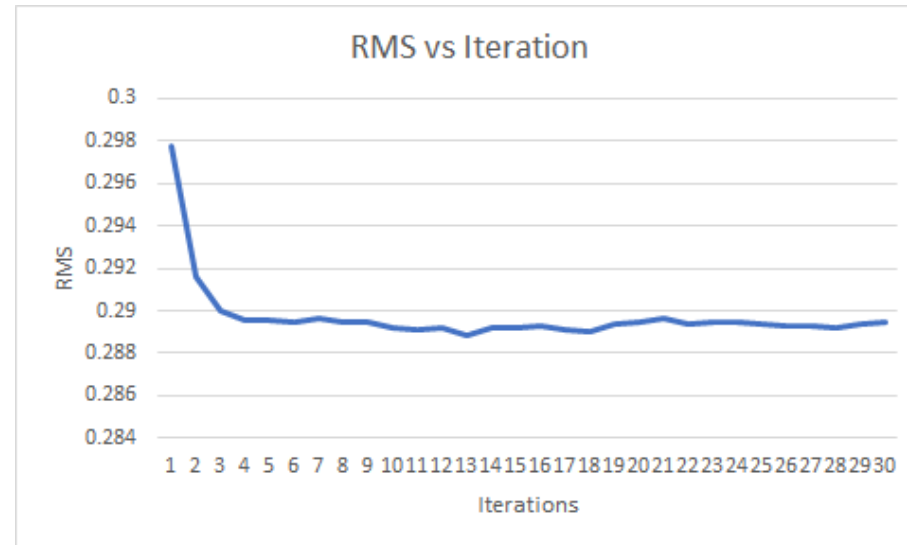
## Reducción de Distorsión Final



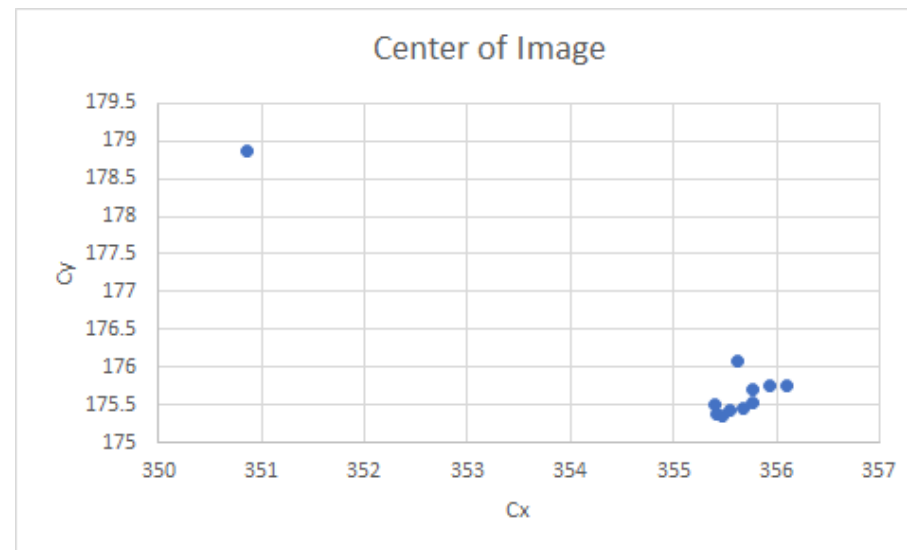
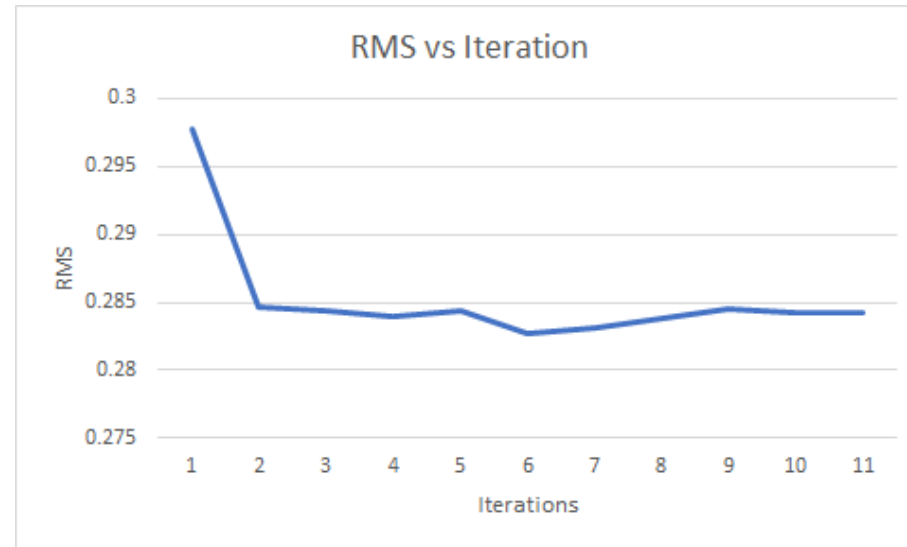


# PlayStation2 Cam

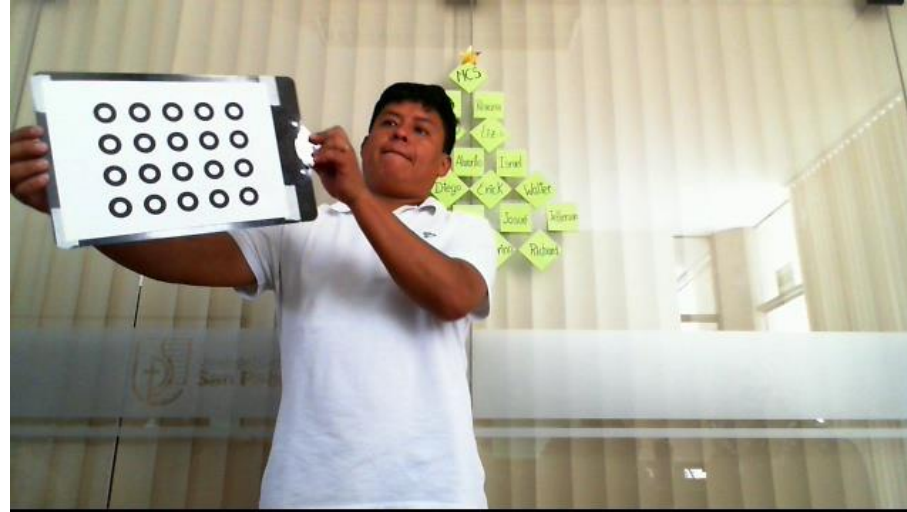
# Algoritmo Iterativo Inicial



# Algoritmo Iterativo Final



## Reducción de Distorsión Final



Gracias