📝 Notas

28 ago 2025

## Visión por Computadora I (1/8) - a213b25

Invitados [mdorogov@fi.uba.ar](mailto:mdorogov@fi.uba.ar) [Visión por Computadora I - CEIA 21Co2025](mailto:vpc1_a21@cursoscapse.com)

Archivos adjuntos [Visión por Computadora I (1/8) - a213b25](https://www.google.com/calendar/event?eid=MHQxaDk3NmdicTljbzlyaDFwcWtrNmVzZ2sgZ29vZ2xlLm1lZXRAbHNlLXBvc2dyYWRvcy5maS51YmEuYXI)

Registros de la reunión [Transcripción](?tab=t.jkd5722mqnu4) [Grabación](https://drive.google.com/file/d/1xYCvsqyuJdOHNYPBONo0r3d8eW3EMGQX/view?usp=drive_web)

### Resumen

Maxim Dorogov dio la bienvenida a la clase de Visión por Computadora Uno, describiendo su experiencia en sistemas de visión artificial y la estructura "tentativa" del curso, que cubrirá desde conceptos básicos con OpenCV hasta algoritmos avanzados y aplicaciones en robótica y detección de objetos. Lse posgrados explicó en detalle la importancia de la iluminación y el hardware en la captura de imágenes, tipos de lentes, sensores de imagen como CCD y CMOS, y el procesamiento de la señal de la cámara. También demostraron la manipulación de imágenes con OpenCV, la conversión de espacios de color como HSB a RGB, y técnicas de segmentación de imágenes usando rangos de color y distribuciones estadísticas, incluyendo un ejemplo con pimientos.

### Detalles

* **Introducción a Visión por Computadora Uno** Maxim Dorogov dio la bienvenida a la materia Visión por Computadora Uno y se presentó como ingeniero electrónico con experiencia en sistemas de visión artificial ([00:00:00](#_f7ykdpqtvy5f)). Mencionó su experiencia laboral en empresas de analítica de retail, analítica deportiva, optimización de redes neuronales para hardware AMD y desarrollo de un framework de optimización de modelos en Intel ([00:01:42](#_fvlhf6eewy4t)). También comentó que ha estado dando clases en el posgrado de la UA por cuatro años y que este año está cursando materias del posgrado como oyente, aprovechando un "hack" que permite a los docentes cursar gratis ([00:03:08](#_2fzfgyv3les4)).
* **Estructura de la Materia y Temas Clave** Maxim Dorogov describió la estructura de la materia como "tentativa" pero aseguró que todos los temas listados se cubrirán, incluyendo temas opcionales que pueden variar de clase ([00:04:24](#_gjfuks3c8228)). Explicó que la materia comienza con conceptos básicos y una introducción a OpenC, el framework más utilizado en visión ([00:05:36](#_ycs14jo6m9r5)). A medida que avanza el curso, se abordarán temas más profundos como histogramas, operaciones a nivel de píxel, filtros digitales (lineales y no lineales) y la transformada de Fourier ([00:06:52](#_u6xod5l693z)).
* **Conceptos Avanzados de Visión por Computadora** El curso progresará hacia la extracción de características, donde los algoritmos producirán coordenadas y vectores que describen puntos interesantes en una imagen. También se cubrirán algoritmos de *clustering* aplicados a características de la imagen para separar regiones y el procesamiento de video, incluyendo algoritmos específicos para video que extraen información temporal ([00:08:08](#_ro146gm0dyfi)). Se mencionó un examen a mitad de curso, similar a un cuestionario, y la importancia de los tres trabajos prácticos (TPs) que engloban todos los temas de la materia ([00:09:33](#_webxqswh6gk5)).
* **Dinámica de Clases y Requerimientos Técnicos** Las clases de Visión por Computadora Uno durarán aproximadamente dos horas y media, combinando teoría (1.5 horas), práctica (30-45 minutos) y un espacio para consultas ([00:10:50](#_ehr5hlc2sugs)). Se requerirá Python, Numbai y OpenCB, con una mención adicional de Supervision para visualización. Los trabajos prácticos y el código de clase se entregarán en Jupyter Notebooks, subidos a un repositorio, y no se aceptarán entregas en Google Colab o archivos comprimidos ([00:12:19](#_rbakukomrt4b)).
* **Bibliografía Sugerida y Flexibilidad de Grupos** Maxim Dorogov sugirió una bibliografía para la materia, destacando un libro que, aunque bueno, está dirigido a quienes ya tienen conocimientos de visión por computadora. Otro libro recomendado explica los conceptos básicos de visión usando exclusivamente OpenCB, lo que lo hace útil para aprender ambos temas simultáneamente ([00:13:27](#_3unfhl1g47df)). En cuanto a los grupos, los estudiantes pueden formarlos con la cantidad de personas que deseen, incluso trabajando solos, y pueden variar la composición de los grupos para cada TP ([00:14:50](#_iuz1q2q5b9t0)).
* **Complejidad de los Algoritmos de Visión por Computadora** Maxim Dorogov explicó que desarrollar algoritmos de visión por computadora es complejo porque, además de emular sistemas cognitivos como otros algoritmos de *machine learning*, deben modelar la percepción humana y la interacción de la luz con el entorno ([00:15:54](#_mz7i4y9qqo8d)). Señaló que para los humanos es fácil interpretar imágenes difuminadas o pixeladas, pero para los algoritmos, estos pueden ser problemas irresolubles ([00:17:16](#_2fm5kebfasli)).
* **Clasificación, Detección y Segmentación de Objetos** Se describieron los problemas de clasificación, detección y segmentación de objetos en visión por computadora ([00:18:28](#_4e2yhfpmwmtz)). La clasificación implica identificar la clase a la que pertenece una imagen, la detección localiza y clasifica objetos dentro de una imagen usando recuadros, y la segmentación asigna una clase a cada píxel de una imagen para crear máscaras detalladas ([00:19:58](#_nckhmsgopvw)). Maxim Dorogov destacó que la dificultad y el costo de los modelos aumentan progresivamente de clasificación a segmentación debido a la complejidad de la anotación de datos ([00:22:36](#_rwp6bkgxfqtu)).
* **Extracción de Características y Aplicaciones en Robótica** Se explicó la extracción de características más abstractas, donde los algoritmos identifican puntos únicos que caracterizan regiones, independientemente de transformaciones como la rotación o el cambio de escala ([00:24:03](#_80on70e8hz6m)). Maxim Dorogov mencionó que esta técnica es crucial en robótica, especialmente en Simultaneous Localization and Mapping (SLAM), donde resolver la correspondencia de puntos ayuda a mapear el entorno y autoposicionarse ([00:25:37](#_69041pzgefcf)). Otro ejemplo incluye el rastreo de trayectorias en deportes al ubicar puntos clave en una superficie de control y transformar coordenadas ([00:27:04](#_kbv8cft0xwb9)).
* **Modelos Generativos y Restauración de Imágenes** Se habló de modelos generativos, como Stable Diffusion, que pueden alterar imágenes y resaltar detalles ([00:29:40](#_1r0od789aool)). Juan Pablo Gonzalez preguntó sobre la restauración de imágenes muy pixeladas, a lo que Maxim Dorogov respondió que, aunque hay herramientas como Google AI Studio que pueden mejorar la resolución de baja a 4K, imágenes extremadamente pixeladas probablemente no se puedan restaurar de manera significativa ([00:36:54](#_culdx36oue4i)). También se mencionó el escalado de resolución con redes neuronales en hardware para televisores y consolas, que ofrece buenos resultados en videojuegos ([00:39:44](#_peacf5lxdcox)).
* **Aplicaciones de la Visión por Computadora en la Vida Cotidiana** Maxim Dorogov resaltó la omnipresencia de los algoritmos de visión hoy en día, mencionando ejemplos como robots de entrega, dispositivos de realidad virtual (Oculus) y sistemas de asistencia en automóviles modernos, incluyendo el monitoreo de líneas de carril, la visión de 360 grados y la lectura de señales de tráfico ([00:39:44](#_peacf5lxdcox)). También hizo una analogía con el Pokédex de 1994, señalando cómo lo que parecía fantasioso entonces es hoy una aplicación básica de la visión por computadora ([00:43:30](#_u2ww9m4322x3)).
* **Historia y Evolución de la Visión por Computadora** Se presentó una línea del tiempo de conceptos relacionados con la visión por computadora, abarcando desde modelos de lenguaje con visión y algoritmos de segmentación hasta cadenas de Markov y filtros de Kalman, algunos con orígenes en la década de 1970 ([00:44:45](#_t0g3xjripwsc)). Se destacó que muchos conceptos de *deep learning*, como las redes convolucionales, surgieron en los 90 o 2000, pero solo pudieron implementarse ampliamente en la última década debido a la falta de hardware adecuado ([00:46:06](#_fatyvar2o32b)).
* **Percepción Humana del Color y la Intensidad** Maxim Dorogov explicó cómo los humanos perciben el entorno, centrándose en las células del ojo: bastones para sombras y texturas en baja luz, y conos para percibir colores (azul, verde, rojo) ([00:47:26](#_xvt0b81im3dr)). La percepción de la intensidad lumínica no es lineal, similar a la percepción auditiva, lo que lleva a la necesidad de algoritmos de corrección gamma en las cámaras para deslinearizar la información del sensor ([00:50:19](#_9o89tcv6rugr)). Se mencionó que el daltonismo es más común en hombres debido a la codificación de las células de color en un cromosoma específico ([00:51:35](#_r0hl0qyxur4o)).
* **Espacios de Color y su Aplicación** Se introdujo el concepto de colores metámeros, que tienen la misma longitud de onda pero se perciben diferente, un problema que llevó a la estandarización del color en 1930 ([00:53:06](#_8j88b2df6gdz)). Se discutieron los espacios de color, como RGB y XYZ, señalando que mientras RGB es cartesiano y se usa en la mayoría de los dispositivos, otros espacios como Cielab o HSB permiten separar los componentes de luz y color, facilitando la manipulación independiente del brillo y el matiz ([00:54:20](#_9uoms6vd9lav)). En OpenCB, las funciones para la conversión de espacios de color son automáticas, siendo los más comunes RGB, HSB y escala de grises ([00:58:40](#_b8mo6aqbnokz)).
* **Conversión de Colores HSB a RGB** Daniela Putrino y lse posgrados discutieron la conversión de colores de HSB a RGB. Lse posgrados explicó que existen tablas para mapear RGB a HSB, y que esta conversión es útil para modificar el brillo sin alterar el color o para hacer seguimiento de objetos donde el RGB podría ser complicado ([01:00:03](#_cjacmist28fa)). También destacaron que blanco y negro tienen el mismo valor H, diferenciándose por el "value" ([01:01:20](#_w99ows6eboni)).
* **Procesamiento de Imágenes y Videos** Lse posgrados introdujo el proceso previo a la escritura de un algoritmo de procesamiento de imágenes o videos, enfatizando que la calidad de la imagen comienza antes de la digitalización. Se mencionó la importancia de la iluminación adecuada y el hardware apropiado para evitar problemas irrecuperables en la captura de imágenes ([01:02:57](#_6ioyxv74yjl9)).
* **Variables de la Cámara** Se discutieron diversas variables importantes en la elección de una cámara, como el tipo de sensor, el obturador (shutter), el tipo de interfaz y las tablas de consulta (LUTs). Lse posgrados mencionó cámaras con sensor de profundidad y cómo la interfaz puede afectar la implementación en entornos de producción ([01:15:01](#_l59yjs5njftj)).
* **Tipos de Lentes e Iluminación** Lse posgrados explicó los diferentes tipos de lentes, como las telecéntricas y las endocéntricas, y cómo afectan la captura de imágenes ([01:16:25](#_y7jz6gms22fg)). También se abordó la importancia de la iluminación, mostrando ejemplos de cómo el ángulo de la luz incide en la visibilidad de los detalles del objeto, como el uso de luz de fondo o campo oscuro ([01:17:40](#_9v5d41jmdoyr)).
* **Procesamiento en la Nube vs. Local** Se diferenció el procesamiento de imágenes en la nube (cloud) del procesamiento en el mismo lugar donde está la cámara (Edge). Lse posgrados señaló que cada enfoque tiene su propia arquitectura y que la elección depende del producto que se quiera desarrollar ([01:17:40](#_9v5d41jmdoyr)).
* **Iluminación Fuera del Espectro Visible** Lse posgrados explicó las ventajas de iluminar con luz fuera del espectro visible (UV o IR), como en cámaras de vigilancia que pueden ver en la oscuridad total. Se demostró cómo esta técnica puede revelar información adicional de los objetos, como en el caso de billetes iluminados con infrarrojo ([01:19:08](#_6qwf35m074vp)).
* **Lentes Telecéntricas y Errores de Perspectiva** Se profundizó en las lentes telecéntricas, que eliminan los errores de perspectiva al permitir solo el paso de haces de luz paralelos al eje óptico. Lse posgrados indicó que estas lentes son cruciales para la metrología, donde la precisión en las mediciones es fundamental ([01:20:25](#_cb2klwpjuelr)).
* **Sensores de Imagen: CCD vs. CMOS** Lse posgrados describió los dos tipos principales de sensores, CCD y CMOS, destacando que CMOS es el estándar actual. Se explicó que los sensores CCD son más sensibles y se usan en aplicaciones específicas como la astrofotografía, mientras que los CMOS ofrecen una señal menos ruidosa al digitalizar la información en cada fotositio ([01:23:00](#_o8gqy1ek1zrl)).
* **Distorsiones en Sensores de Imagen** Se abordaron problemas como el "blooming" y el "smearing" en sensores CCD, donde una luz intensa puede desbordar la carga a píxeles vecinos. Lse posgrados también explicó el "rolling shutter" en sensores CMOS, que puede causar distorsiones en objetos en movimiento rápido debido a la lectura por filas o columnas ([01:24:31](#_jq46f1iu2sm6)).
* **Clasificaciones Adicionales de Cámaras** Lse posgrados mencionó otras clasificaciones de cámaras más allá del sensor, como réflex o mirrorless, y full frame o no full frame, que dependen más de la geometría de la cámara. Se habló sobre el uso de microlentes para compensar la menor área fotosensible en algunos sensores y para el filtrado de color en el arreglo de Bayer ([01:25:42](#_2pvftdwjoe6v)).
* **Procesamiento de la Señal del Sensor** Se describió el pipeline de una cámara, incluyendo la lente, el control de apertura (iris) y el sensor. Jose Luis Diaz y Juan Pablo Gonzalez confirmaron que el ISO amplifica la señal del sensor, pero también el ruido, lo que explica el "grano" en las imágenes con ISO alto ([01:28:31](#_rzscyhwzwz4k)).
* **Filtros y Compresión de Imágenes** Lse posgrados explicó los filtros aplicados después de la conversión digital, como el filtro de mosaico para compensar los artefactos del arreglo de Bayer y algoritmos para enfocar y reducir el ruido. También se mencionó el balance de blancos, la curva gama para el contraste y la compresión de imágenes a formatos como PNG o JPG, que reducen el rango dinámico ([01:30:47](#_g273wast5txw)).
* **Representación Digital de Imágenes** Se definió una imagen digital como una matriz de n por m por 1 (escala de grises) o n por m por 3 (color). Lse posgrados aclaró que los valores de píxel van de 0 a 255 (unsigned int), donde 0 es negro y 255 es blanco, y que en OpenCV se indexa como una matriz ([01:32:07](#_1tcy57fcvm2i)).
* **Nomenclatura de Resolución y Dimensiones** Lse posgrados hizo una aclaración importante sobre la nomenclatura de la resolución y las dimensiones de la matriz de una imagen. Aunque se dice que un monitor tiene, por ejemplo, 1280x1024 (ancho por alto), al levantar una imagen como una matriz en OpenCV, las dimensiones se invierten, siendo 1024x1280 (filas por columnas) ([01:33:30](#_tlmgrysk46dg)).
* **Práctica de Procesamiento de Imágenes con OpenCV** Lse posgrados introdujo una práctica con notebooks de Python usando NumPy, OpenCV y Matplotlib para la manipulación de imágenes. Demostró cómo leer una imagen en escala de grises con `imread` de OpenCV y cómo visualizarla con `imshow` de Matplotlib, explicando la necesidad de especificar el mapa de color para una visualización correcta en escala de grises ([01:54:19](#_z32y1aurni7i)).
* **Extracción de Información de la Imagen** Lse posgrados mostró cómo graficar el perfil de una columna de píxeles para extraer información de la imagen, como los niveles de intensidad y la ubicación de objetos ([01:57:00](#_8c4vowl0cczn)). Explicó que al analizar los niveles de intensidad se puede determinar información del entorno, como la altura de la cámara y el tamaño del sensor ([01:58:22](#_hxz04vq6vgfg)).
* **Procesamiento y visualización de imágenes** lse posgrados explicó que las imágenes pueden procesarse columna por columna, donde cada columna se representa como una matriz de números que representan los píxeles. lse posgrados demostró que las imágenes pueden modificarse, por ejemplo, rellenando una columna con ceros para crear una línea negra ([01:59:33](#_ygjvklml3a3e)). También explicó que OpenCV usa el formato BGR para el color, no RGB, lo que afecta la visualización de la imagen, pero se puede convertir con la función `convert color` para una correcta visualización con Matplotlib ([02:00:47](#_ro1736l9t349)). lse posgrados también mencionó que OpenCV proporciona funciones para desarmar y fusionar canales de color, lo que permite el procesamiento individual de la información de cada canal ([02:02:22](#_r157n6s9ezj8)).
* **Análisis y manipulación del color en imágenes** lse posgrados demostró cómo analizar los perfiles de color en una imagen a color, notando que, aunque los valores numéricos de cada canal son distintos, muestran una correlación ([02:03:38](#_7rmgg9ajehhc)). Explicó que el espacio de color RGB no es ideal para trabajar debido a esta correlación y a la redundancia de información que introduce. También se explicó que `IM WRght` permite guardar imágenes en diversos formatos, ya que OpenCV se encarga de seleccionar el códec apropiado ([02:04:57](#_c6z092fq76i3)). Daniela Putrino preguntó sobre los formatos de entrada, a lo que lse posgrados respondió que pueden ser cualquiera, ya que OpenCV utiliza FFmpeg ([02:06:25](#_l6jxieg7ckhb)).
* **Anotaciones en imágenes y segmentación** lse posgrados demostró cómo añadir anotaciones a las imágenes usando funciones de OpenCV para dibujar líneas, rectángulos y texto ([02:07:41](#_4db34i4lpvpf)). También explicó cómo ajustar el tamaño del texto para crear un fondo, lo que mejora la legibilidad. lse posgrados presentó el desafío de segmentar una mesa de forma automática, independientemente de la posición de la cámara, utilizando espacios de color y estadística ([02:08:56](#_idup3nspo52u)). Explicó que el espacio de color HSB es útil para la segmentación, ya que el Hue se mantiene consistente en la mesa, a pesar de las variaciones de tono ([02:11:51](#_szo5x8sovx4y)).
* **Técnicas avanzadas de segmentación y restauración de imágenes** lse posgrados demostró que, al muestrear el color de la mesa en HSB, el valor de Hue se mantiene constante, lo que permite segmentar la mesa utilizando un rango de valores de Hue ([02:11:51](#_szo5x8sovx4y)). Con la función `in range`, se puede definir un rango de Hue para identificar los píxeles de la mesa, lo que resulta en una segmentación efectiva, aunque puede ser ruidosa inicialmente ([02:13:23](#_2mnwo1h0ah8g)). Daniela Putrino preguntó sobre la posibilidad de obtener un `unique` de los valores de Hue para identificar las medias de cada objeto, a lo que lse posgrados sugirió usar un histograma del Hue y trabajar con `Bans` para acotar la solución ([02:15:53](#_dnaf85dh7qdy)). lse posgrados también explicó que el rango de Hue en OpenCV es de 0 a 180, lo que difiere de la teoría de 0 a 360, y destacó que esta técnica es elemental y funciona mejor en objetos con colores homogéneos y sin repeticiones de color en la imagen ([02:17:09](#_putkutjhyt7e)).
* **Segmentación de pimientos y distribución de colores** lse posgrados mostró un método de segmentación más avanzado para un pimiento amarillo, que tiene variaciones de color y requiere tomar múltiples muestras ([02:18:15](#_d8xaw14mmzzh)). lse posgrados explicó que se tomará la media y la desviación estándar del color en el espacio RGB para cada canal. Luego, se identificarán los píxeles dentro de un rango de seis sigmas para capturar la mayor parte del espacio muestral, lo que es óptimo si la distribución de color es gaussiana ([02:19:38](#_jps8h4z73r2x)). lse posgrados mostró que la aplicación de esta máscara usando `bitbice ant` resulta en la segmentación del pimiento, aunque algunas partes como el tallo o regiones con verde no se segmentan, ya que están fuera del rango de seis sigmas ([02:20:56](#_tlr29gr0wehq)). Daniela Putrino preguntó cómo se le indicó al sistema que se estaba mirando el pimiento amarillo, y lse posgrados respondió que se hizo recortando un pedazo específico de la imagen con coordenadas predefinidas ([02:22:07](#_17q7fw5ax9d5)).
* **Distribución de colores y uso de IA** lse posgrados explicó que la distribución de colores en RGB no es gaussiana, lo que dificulta el uso de la media y la desviación estándar para la segmentación. Sin embargo, al graficar la distribución en HSB, los canales son más "clusterizables" y el amarillo se concentra en un rango específico de Hue, lo que permite segmentar el pimiento amarillo de manera más efectiva ([02:22:07](#_17q7fw5ax9d5)). Los tallos verdes y los brillos no se segmentan porque sus Hues están fuera del rango seleccionado ([02:23:37](#_79brs6u3nj66)). lse posgrados también compartió experimentos de restauración de imágenes utilizando herramientas de IA como WHK de Google y Gemini, que no solo mejoraron la calidad de la imagen, sino que también interpretaron el contenido y generaron videos a partir de imágenes estáticas ([02:25:07](#_n8li13sxsc0l)). Finalmente, Ricardo Silvera preguntó sobre la particularidad del color verde en el croma, y lse posgrados explicó que se usa un color que no se repita en el sujeto, siendo el verde flúor una solución estandarizada por conveniencia .

### Pasos siguientes recomendados

* lse posgrados enviará un correo electrónico con el repositorio donde se encuentra todo el material del curso después de la clase.

*Revisa las notas de Gemini para asegurarte de que sean correctas.* [*Obtén consejos y descubre cómo toma notas Gemini*](https://support.google.com/meet/answer/14754931)

*Danos tu opinión sobre el uso de Gemini para tomar notas en una* [*breve encuesta.*](https://google.qualtrics.com/jfe/form/SV_9vK3UZEaIQKKE7A?confid=_UDeaCCIk001BVcIhc6uDxIQOAIIigIgABgFCA&detailid=unspecified)

📖 Transcripción

28 ago 2025

## Visión por Computadora I (1/8) - a213b25 - Transcripción

### 00:00:00

**lse posgrados:** Buenas, ¿cómo andan? ¿Todo bien?  
**Jose Aviani:** Buenas, ¿qué tal? ¿Todo bien?  
**lse posgrados:** Bueno, perdón por la por la demora, pero que se me se me desbloqueó se me desbloqueó la la compu que uso para para presentar y puse mal el código de verificación que te pide Google y tuve que empezar todo todo de vuelta. un quilombo que solemos arrangar eh a horario. Bien, bueno, eh nada, bienvenidos a a Visión por computadora uno. Eh, no sé si alguien tiene alguna consulta administrativa o algo así antes de de que arranquemos así formalmente con con la materia, si alguien que le esté haciendo oyente, alguien que no sé, no tiene la correlativa o algo por el estilo. Si no, arrancamos, arrancamos no más. Bueno, eh, nada, mi nombre es Maxim Dorogov. Sí. Eh, estudié y me recibí sorprendentemente acá en la UA. Sí, soy ingeniero ingeniero electrónico. Eh, desde hace unos 4 años que estoy dando clase acá en el posgrado. Sí, desde hace unos seis que laboro en sistemas de visión artificial. En verdad digo seis porque hace seis me recibí general.  
 

### 00:01:42

**lse posgrados:** Uno cuando trabajó a lo largo de toda la carrera tiene más años de experiencia, pero digo, tengo seis porque nada, los seis, hace seis obtuve mi primer laburo como ingeniero, digamos, para una empresa. Eh, nada, en cuanto en cuanto a laboral, sí, actualmente trabajo en una empresa que hace analítica de retail, ¿sí? O sea, es medio medio un embole, pero está bueno. Se analizan datos demográficos de clientes que entran a negocios a través de de procesamiento de de imágenes, procesamiento de video. Hay varias cosas, hay interesantes, hay modelos de lenguaje, hay mezclas de agentes, hay algoritmos clásicos, hay mucho de por ahí OpenCB, no sé si seguramente algo habrán visto en el posgrado. como un rejunte de todo lo que uno puede usar en un sistema de de machine learning. Eh, nada, es una empresa de que hoy está en Dubai y todos los clientes son de Dubai, con lo cual hay como ciertas cosas eh que por ahí en un entorno occidental no se dan. Sí. E previo a eso, eh, que dice, "Bueno, elaboré para una empresa en Boston que hacíamos analítica deportiva." Ahí estaba bastante bueno este porque era medio medio imposible lo que querían lo que querían lograr.  
 

### 00:03:08

**lse posgrados:** Previo a eso lauré en AMD a través de una consultora uruguaya. Bueno, ahí hacíamos optimización de redes neuronales. Yo siempre el lado de visión por computadora para hardware de AMD, digamos, sistemas de superresolución de escalado y tipo de cosas más relacionadas a a la parte gráfica. Y previa a eso laaburé en Intel eh esta vez para para Intel Posta sin una consultora, ¿sí? En en Oreon. Y ahí eh forma parte del equipo que desarrolló un framework que se llama Open Vino. Es un framework de optimización de de modelos también. Eh, así que nada, en cuanto a lo a lo personal, creo que ya les comenté, vengo dando clase acá en el posgrado, doy clase en otra materia también de de la facción a visión, pero eh por ahí un poco más matemática, sí, como toda materia de grado de de la UA. Eh, y bueno, desde hace este año estoy cursando también algunas materias del posgrado. Seguramente hemos compartido algún algún curso. Estoy haciendo casi todo oyente porque hay como un hack ahí que si vos sos docente eh, podés cursarlo gratis. Yo me vivé recién ahora el cuarto año de ser docente acá.  
 

### 00:04:24

**lse posgrados:** Y el hack es que si vos metés materias los días que das clase, tenés la falta justificada, obviamente, y podés nada por ahí meter que son cinco bimestres en tres, ¿sí? Porque metés toda la materia los jueves, decís, "No, yo doy clase." Y después tenés toda la semana para ver los videos en por dos y entregas los PPS y ya está. Así que nada, es un un trucazo, la verdad. Bueno, eh, ¿das, consultas de lo que comenté? Si no vamos a ir avanzando. La materia tiene esta estructura que es medio tentativa en realidad. Sí. Eh, a grandes rasgos, todos los temas que están listados acá se dan, pero eh hay temas opcionales que se van dando. De hecho, acá dice teórica opcional que figura en la clase ocho después del examen, pero no sé, la corte anterior la dimos en la clase seis, que fue un poquito más corta de lo normal y otra veces le damos en la clase siete, que si bien es una clase medio pesadita, es más corta, ¿sí? Eh, a diferencia de todas las demás que suelen durar hasta las 9:30, 10 en función de cómo vengan con las preguntas y y todo eso.  
 

### 00:05:36

**lse posgrados:** Es arrancar, seguir con conceptos básicos. Esta es la primera materia que uno que uno tiene de visión por computadora y tampoco es un posgrado de visión por computadora. Entonces, nada, seguramente eh hay temas que no los tocábamos con la profundidad que a mí me gustaría, pero sí la idea es tratar de abarcar todo o por lo menos todo lo que mi experiencia y en la experiencia de otras personas con las que suelo trabajar, eh, digamos, temas que se usen, sí, no necesariamente temas que haya usado yo, pero sí temas que se usen en la industria en general. E el que ya está laburando en visión por computadora, seguramente las primeras clases se va a pegar un embole. Sí, pues no vamos a ver nada sí nada muy novedoso. Vamos a arrancar con cosas básicas, ¿sí? Vamos a ver conceptos de visión, esto la clase uno sea hoy y vamos a tener una introducción a lo que sería OpenC, que es el framework eh que se suele usar en el 99% de las veces para hacer algoritmos de de visión. A medida que que vayamos avanzando acá con con los temas, nos vamos a ir abstrayendo un poco más, ¿sí?, de lo que es una imagen. Sí, de hecho, vamos a ver que ya en la clase dos tenemos cosas.  
 

### 00:06:52

**lse posgrados:** Esto sí seguramente lo vieron, ¿saben lo que es un histograma por ahí? Eh, no necesariamente un histograma como el de probabilidad y estadística, sino el histograma que te tira una cámara cuando vos ponés el modo histograma, que puedes ver ahí la distribución de de grises en tu en tu pantalla. Sí, son operaciones a nivel píxel, sí contamos valores de píxeles, aplicamos funciones a píxeles, ¿sí? O sea, estamos modificando una imagen y a medida que que van avanzando las clases, vamos a seguir viendo temas por ahí un poco más profundos. Ya en la clase tres vamos a ver eh filtros digitales, ¿sí? Vamos a trabajar tanto con filtros lineales como no lineales. Y vamos a ver un tema que en sí no es un tema de procesamiento de imágenes, pero es una gran herramienta matemática, ¿sí?, que es la transformada, que nos permite por un lado hacer cosas que en el dominio eh espacial de la imagen no podríamos y por otro permite acelerar ciertas cosas, sí, que capaz haciéndolas con la imagen así como la conocemos y tardamos el trip. Eh, hasta acá nuestro input va a ser una imagen, nuestro output va a ser una imagen procesada. ¿Sí? Y de acá en adelante ahí sí nos empezamos a abstraer bastante más.  
 

### 00:08:08

**lse posgrados:** Sí, por ejemplo, cuando hagamos extracción de características, vamos a entrar con una imagen a nuestro algoritmo y vamos a salir con coordenadas Xi de puntos que según el algoritmo son puntos interesantes para la imagen y un vector que describe esos puntos como si fuera un embeding que caracteriza un punto Xi en particular. Sí, en la clas se nos metemos a eh algoritmos de clustering, pero aplicados a las características de la imagen para poder separar regiones a partir de eh bueno, las características de de esas de esas regiones. Sí. Y en la clase si, aunque en realidad ya en el TP2, que lo vamos a ver dentro de un par de clases, eh nos metemos con procesamiento de video. Sí, vamos a ver algoritmos por en sí un video está compuesto por una sucesión de de imágenes, con lo cual todo lo que vemos de acá para arriba lo podemos aplicar. De hecho, lo van a hacer para uno de los trabajos prácticos, pero hay ciertos algoritmos que sirven para procesar video propiamente dicho. Sí. y que bueno sirven para extraer información que nos da un video y que por ahí una imagen o una sucesión de imágenes, pero que no está correlacionada entre sí, no no no nos la no nos la provee. Sí. Y esto puede ser que cambie, puede ser que no.  
 

### 00:09:33

**lse posgrados:** Sí. eh en función de, no sé, si nos toca algún feriado o algo por el estilo, este año venimos venimos muy bien porque siempre uno ahí por ir se enferma o algo y se enferma los jueves. La clase 8 se suele tomar un examen que es comparable a un, no sé, a un cuestionario de del campus, sí, que seguramente tuvieron en alguna otra materia, pero nada, con la con la ventaja de que nos conectamos todos al horario de siempre, ponemos una hora, pueden hacer consultas en el medio, ¿no? Por si hay alguna duda. Y la nota de elecciones se promedia con la nota de los tres trabajos prácticos, ¿sí? con lo cual es raro que alguien desapruebe el examen. Sí, digamos, si siguieron la materia y hicieron los TPS, en general la nota del examen suele ser el promedio de los TPS, sorprendentemente. Sí. Eh, pero nada, está bueno por ahí tomarlo de forma eh sincrónica, cosa de que bueno, alguien por ahí se pueda se pueda sacar una duda en vivo y eventualmente tenemos una teórica extra que no suele entrar. No importa si le damos la clase ocho o si le damos la clase seis, sí, son temas que por ahí escapan al contenido de la materia, pero que están buenos, está bueno saberlos.  
 

### 00:10:50

**lse posgrados:** Eh, tenemos tres TPS, ¿sí?, eh, que engloban todos los temas de la materia. De hecho, el el TP3 se da antes de la última clase, bastante antes, pero lo pueden resolver de varias formas distintas, ¿si? y algunas de esas formas por ahí la vemos eh una o dos clases después de de dar el el TP. En cuanto a la dinámica de las clases, en total sí más o menos sumamos una hora y media de teoría con 10, 15 minutos de descanso cada vez que terminamos una una como una unidad temática y eh más o menos media hora, 45 minutos de práctica, ¿sí? metidos entre los bloques de teoría que vamos viendo, salvo por ahí la clase de hoy, donde la teoría y la práctica están un poco descorrelacionadas. Sí, la idea de la clase teórica es ver conceptos que nos llevan a hacer un algoritmo y en la práctica vamos a ver eh algunas funciones de OpenCB, ¿sí? Y nada, y los paquetes que vamos a estar eh usando. Es tener más o menos eh 2 horas y media de clase, cosa de que quede siempre un espacio de consultas para los trabajos prácticos. Eh, por ahí la clase de hoy, eh, y tal vez la que sigue la cortemos un poquitito antes porque, bueno, no hay muchas consultas sobre TPS porque el primer TP lo vamos a tener la clase que viene, pero bueno, el espacio eh está.  
 

### 00:12:19

**lse posgrados:** En cuanto a los paquetes, no hay mucha ciencia. Sí, cualquier versión de Python, eh cualquier versión de Numbai, eh prácticamente cualquier versión de OpenCB. Sí, acá me falta agregar eh Supervision, que es otro paquete de visualización que vamos a usar en algunos notebooks. Eh el colap, yo pensé que lo había tachado cuando edité la la DIAP. Esto no va. Sí. Eh, los TPS y la el código de de cada clase, sí, están Júpiter Notebook. La idea de los trabajos prácticos es que eh se entreguen en Júpiter Notebook también, ¿sí? O sea, lo suben a un repo, me mandan el link y esa sería la entrega. No, no acepto entregas en Google Colab porque no lo puedo ejecutar y tampoco me manden el código en un en un zip porque es un quilombo eso. Yo prefiero que lo suban un repo, me comparten el link o me invitan al repo si es privado y ahí lo lo puedo lo puedo corregir. Esto de Anaconda es opcional, ¿no? con Sarconda, Miniconda, eh el el gestor de environment nativo de Python también. Sí, es más que nada como una sugerencia, ¿sí?  
 

### 00:13:27

**lse posgrados:** para que no se mezclen con con los paquetes. Hay una bibliografía, sí, que es eh eso sugerida. En realidad sí, en sí toda la materia es como autoconclusiva, tanto la práctica como la teórica, pero un libro que está muy bueno es este de acá, lo pueden bajar gratis de la página del autor. Eh, es un libro que está muy bueno, es un como un buen libro de consulta, pero eh como todo buen libro de de Computer Vision, está pensado para gente que ya sabe el Computer Vision. Entonces, por si uno dice, "Che, lo voy a leer para aprender." Y capaz que no, o sea, es como tenés que aprender antes de leer el libro y cuando ya aprendiste podés ver el libro y entender algunas cosas. En ese sentido está está bastante viola. Y después así como en orden de creciente de importancia, ¿sí? Están todos estos. Este, si nunca programaron nada en su vida o no sé, no entienden cómo usar OpenC, este es un buen libro porque es un libro de computer vision, de conceptos básicos, pero los explica usando Open CB exclusivamente. Entonces como que pu pueden aprender las dos cosas, pero en lo personal yo me quedaría con este a modo de consulta y miraría este solo por curiosidad.  
 

### 00:14:50

**lse posgrados:** Eh, nada más. Bueno, dudas, consultas de lo que de lo que comenté. Los grupos los pueden armar de la cantidad de personas que quieran. en sí, en general creo que cuatro es un número óptimo. Eh, lo pueden hacer solos si quieren. Eso la verdad que es es indistinto. Los grupos también los los manejan. Ustedes pueden hacer un TP en grupo, otro hacerlo solo. Sí. O sea, cuando hagan la entrega me mandan el mail, dicen quiénes están en el grupo y yo lo lo corrijo. Eh, por ahí mañana o hoy más tarde, cuando termine la clase, yo les mando un mail con el repositorio donde está todo el todo el material de la materia. Eh, se va actualizando clase a clase. De hecho, recién hice ahí un pequeño edit en una de las diapositivas, eh, con lo cual nada, cada vez que edite algo, yo se los comento en el mail, porque si entran ahora van a ver que están todas las todas las prácticas y todas las teóricas son más o menos las mismas, no es que hay cambios drásticos entre una cursada y otra, pero sí algunos conceptos pueden llegar a cambiar.  
 

### 00:15:54

**lse posgrados:** Digamos, me gusta, me gusta ir actualizándolo aunque sea de a poco para que no sean todas las clases eh iguales. Eh, no, no, el campus no está el campus y lo tengo como un pendiente, pero no no lo vamos a usar por ahora. Eh, bueno, en sí hacer algoritmos de visión por computadora es por ahí una de las cosas más complejas que uno puede tener dentro de lo que es el campo, ¿no?, de de machine learning o o aprendizaje automático, porque para empezar engloba muchas cosas, ¿sí? y las mismas que se engloban en eh cualquier área relacionada a datos tabulares, por ejemplo. Eh, pero tiene que emular algo que por ahí otros algoritmos no. Sí. Otras áreas no. Sí. En general, ¿qué hace uno con un modelo de de machine learning? trata de emular eh bueno, el sistema el sistema cognitivo de una persona, ¿no? Se ha dado un anális dado un una cierta cantidad de información, quiero poder interpretarla, quiero tratar de sacar relaciones de esa de esa información, pero los datos ya están, digamos, los tengo yo en una tabla, simplemente tengo que ver que cómo los pre o postproceso. En visión por computadora está todo eso.  
 

### 00:17:16

**lse posgrados:** Si, de hecho, muchas veces eh uno simplemente usa videos o imágenes para extraer datos y después aplica herramientas que vieron en otras materias para sacar información de esos de esos datos. Pero hay algo que por ahí no está en otras áreas y es que además de emular el sistema cognitivo, tengo que poder emular o modelar la percepción humana, ¿sí? O sea, la forma en la cual la luz interactúa con el medio y nosotros lo podemos percibir. Acá hay un par de ejemplos, ¿sí?, que para nosotros son bastante bastante simples. Sí, si yo les pregunto qué estamos viendo acá, van a decir, "Bueno, acá hay una persona que tiene pinta de estar en una oficina. Acá parece tiene un reloj en la mano. Eh, puede estar hablando por teléfono o se puede estar agarrando la cabeza. Tiene un escritorio, tiene un teclado, tiene un mouse. Sí. Digamos, si uno ya el hecho de darse cuenta que esto es una persona es un montón, ¿sí? O sea, un algoritmo de visión. Eh, ni siquiera una red de estado del arte hoy en día recibe esta imagen y dudo mucho que que pueda detectar algo de de acá, pero para nosotros es una pavada, ¿sí?  
 

### 00:18:28

**lse posgrados:** Y es una imagen que está bastante difuminada. Algo parecido se da en la imagen de acá. Sí, es una imagen con una resolución muy muy baja, está recontrapixelada, eh, pero puedo empezar a extraer un montón de de información. Sé que esto es una bandera, acá hay cuatro personas, sí puedo distinguir un poco la raza de cada de cada persona, sí puedo intuir en dónde están por el tipo de de bandera que estoy viendo. Sí. Eh, pero bueno, son cosas que para nosotros son muy fáciles y las hacemos en tiempo real, prácticamente sin tener un contexto previo de qué es lo que vamos a ver, pero para una una red neuronal o para cualquier algoritmo, no necesariamente una red neuronal, estos pueden ser dos problemas completamente irresolubles, ¿sí? Y simplemente son dos imágenes un poco un poco borrosas, una borrosa y otra un poquito más eh pixelada. ¿Y qué podemos resolver con algoritmos de de visión? Acá hay un ejempload que no está, pero porque es muy muy trivial. Sí. Eh, el ejemplo que no está es el ejemplo de clasificación, ¿sí? O sea, dada una imagen y un modelo que puede diferenciar entre clases, quiero saber si esa imagen pertenece a la clase con la cual yo entré en el modelo o o a las clases con las cuales yo entré en el modelo.  
 

### 00:19:58

**lse posgrados:** Eh, el siguiente ejemplo, sí, que ya es un poco más un poco más complicado es el de detección, donde dado una imagen yo te digo dónde están ubicados los objetos y a qué clase pertenecen. Por ejemplo, armar un dataset de clasificación para entrenar un algoritmo es una apagada, es poner imágenes en carpetas y leerlas. Si le pongo el nombre de la clase a cada carpeta, ya está. Un dataset de detección ya es un poco más complejo porque necesito, bueno, las imágenes guardadas en algún lado y después necesito archivos que describen la metadata que está en es en esas imágenes. Sí. O sea, por cada para cada objeto yo necesito tener escrito con palabras una clase, ¿sí? Necesito las coordenadas Xi, lo cual ya, o sea, las coordenadas Xi que modelan el rectángulo de donde está el objeto, lo cual ya se vuelve engorroso para mantener esas coordenadas X necesito una estructura de datos que mi código pueda interpretar. ¿Sí? Entonces, para cada imagen, yo puedo tener, esto es uno de los posibles formatos, un archivo que puede ser un XML, un Jon o sea un txt con los valores así uno al lado del otro que contenga toda esta información, ¿sí? Para que de alguna manera el modelo lo pueda interpretar.  
 

### 00:21:19

**lse posgrados:** Eh, pero si prestamos atención, vemos que esto está bien, pero hasta ahí, porque, por ejemplo, adentro del rectángulo que dice caballo, eh, hay cosas que no son caballo. Sí. Por ejemplo, hay tierra, hay pasto, hay árboles, hay partes de persona. Sí. Y hay muchos otros objetos también. Por ahí se hace más evidente. Acá donde yo tengo un rectángulo que dice barco, pero adentro del barco hay una, dos, tres, cuatro personas, ya que no es que cada píxel que yo tengo acá dentro de este rectángulo se va a corresponder con un barco. Capaz algunos sí, pero después hay cosas que son personas que están adentro del mismo onbox. Entonces ahí aparecen los algoritmos de segmentación que lo que hacen es una clasificación pero a nivel píxel. O sea, cada píxel dentro de mi imagen le asignan una clase. Una manera de representarlo es, por ejemplo, eh pintando todos los píxeles de la misma clase con un color común. Entonces, todo lo que es eh calle o camino está pintado con gris, todo lo que es persona está pintado con rosa, todo lo que es árbol está pintado con verde. Y se forman lo que uno denomina máscaras de eh segmentación.  
 

### 00:22:36

**lse posgrados:** Sí, esto es mucho más preciso, pero si yo antes para cada imagen de un dataset de detección tenía que armar una estructura de archivos para mantener las coordenadas y bueno, esto es más complicado todavía porque yo puedo tener un formato, por ejemplo, es para cada imagen. Tengo una estructura de archivos con las coordenadas de los contornos de la máscara, lo cual es mucha más información. o para cada imagen puedo tener múltiples imágenes donde cada una de esas múltiples imágenes tiene todo en negro y resaltado solamente la clase que que representa. Sí, armar estos dataset también se vuele cada vez más complicado porque en clasificación lo único que necesito es meter imágenes en carpetas. En detección necesito dibujar rectángulos con algún software. Hay un montón Sebat es uno de los más conocidos. Después está Rflow que te permiten etiquetar imágenes y generan esas estructuras de archivos que uno después levanta. Eh, en segmentación es más complicado porque no es solo dibujar un rectángulo, sino que tengo que pintar la imagen a nivel píxel, ¿sí? Para poder encerrar el objeto ahí. Después, si bien esas software tienen eh tipo como herramientas de asistencia eh y por ahí es más tiempo lo que uno tarda en corregir el algoritmo de segmentación que viene ahí que segmentando voz a mano.  
 

### 00:24:03

**lse posgrados:** Sí, pero bueno, digamos como que la dificultad para entrenar un modelo y armar un dataset va creciendo, ¿sí? Si uno parte de clasificación, pasa a detección y después a a segmentación de de objetos. Lo mismo también eh el costo de de los modelos, ¿sí? El costo en términos de eh uso de memoria y velocidad. Si comparamos dos modelos con una performance similar en detección y en y en segmentación y el modelo de segmentación eh tarda más, sí es mucho más pesado que un modelo de de detección. También podemos hacer detección o mejor dicho hagáis extracción de características, ¿sí? Pero ya hablando de características más abstractas, ¿sí? No necesariamente decir, "Che, esto es un objeto en concreto," sino que tengo algoritmos que detectan puntos que dentro de una región, por ejemplo, acá sobre la señal, son puntos que caracterizan esa región eh de una forma única, lo cual me permite, por ejemplo, detectar estos puntos en otra instancia del mismo objeto. A pesar de que el objeto sea distinto, haya cambiado la perspectiva, esté rotado, si digamos son puntos que son invariantes a cualquier transformación. Acá no solo no solo que es otra instancia del objeto, sino que la calidad de la de la imagen es distinta.  
 

### 00:25:37

**lse posgrados:** Sí. Esta imagen se ve claramente que tiene otra otra tonalidad. Sí. Y eh no solo hay diferencia de escala, sino que el objeto está rotado. Hay una una transformación a fino, una homografía que se le aplicó. Sí, pero así todo si seguimos los puntos más o menos están bastante bastante acertados. Esto se usa mucho en robótica, en particular en lo que es Slam, que es simultanes localization mapping. Porque el algoritmo que uno tiene que resolver para localizar un objeto, digamos, para saber cómo se desplazó del punto A al punto B, eh termina siendo el mismo que uno necesita para mapear el entorno que estoy viendo. Entonces, si uno lo resuelve, resuelve esta correspondencia una vez tiene las dos cosas, mapeó el entorno y a la vez se autoposicionó a términos relativos, ¿sí?, de las cosas que estoy viendo a mi alrededor. Esto más adelante lo vamos a lo vamos a ver. Sí, de hecho creo que en la clase 5 cuando vemos Shift, sí tenemos un par de ejemplos de cómo armar matrices de transformación que nos permiten llevar de este plano a a este plano y y viceversa. ¿Sí? Todo esto a partir de eh el análisis de estas de estas correspondencias.  
 

### 00:27:04

**lse posgrados:** Eh, en el ejemplo anterior la idea era buscar puntos eh abstractos, ¿no? O sea, puntos que caracterizan la imagen, pero que no tienen una correspondencia semántica con la imagen que yo estoy viendo. Podemos tener casos así donde yo quiero ubicar puntos en concreto. Sí, tengo una superficie de control con una determinada perspectiva y tengo lo que está viendo la cámara. Entonces, la cámara lo que hace es decir, "Che, bueno, yo voy a ubicar estos puntos que siempre están en estas ubicaciones, no en píxeles, pero sí en el mundo en el mundo real. Pues si la cámara, no sé, sube, baja, la coordenada del punto en píxeles va a cambiar, se va a alejar o se va a acercar, pero en el mundo real siempre van a estar en el mismo lugar. Eh, y acá sí es donde uno dice, "Che, ¿de qué me sirve esto?" Y yo puedo trazar la trayectoria de del bateador y la velocidad. Sí, este, no, cualquiera lo que no corre hasta acá el tipo en realidad así entre las bases e a través de una una transformación. Esa transformación la logro buscando correspondencias de este plano y este otro plano. ¿Sí?  
 

### 00:28:16

**lse posgrados:** Después lo que hago es multiplicar las coordenadas de la persona por matriz de transformación y puedo ir de este plano de imagen a este plano de imagen. ¿Sí? y puedo trazar todas las trayectorias, puedo calcular velocidades de una forma mucho más eh mucho más cómoda. También puedo ir al revés, o sea, puedo mapear de la superficie de control a el plano de la imagen. Entonces, puedo saber si la persona estaba bien ubicadas o no sé, por ahí acá en en béisbol, la verdad que no se me ocurre, pero si estuviéramos viendo un una proyección de una cancha de basket, podría usarlo para ver si el tipo tiró un triple o un doble en función de cómo estaban la posición de los pies respecto de de la línea de de tres tiros, por ejemplo. Sí. Y fíjense que acá hay líneas que coinciden casi a la perfección y hay otras que que no esto se hizo con datos reales, digamos. O sea, es un screenshot del framework que usábamos en el en el lauro. Eh, por ejemplo, esto acá coincide perfectamente, pero hay otras líneas que tienen cierto offset y acá se pueden dar varias cosas. Obviamente todos los algoritmos tienen cierto cierto error. Este en particular lo que me dice es que la cancha tiene las dimensiones correctas.  
 

### 00:29:40

**lse posgrados:** Sí, porque esto calza perfecto. Pero la los rectángulos de los bateadores los dibujaron medio medio mal. Sí. Eh, lo que sí dibujaron bien y se ve que lo midieron porque, bueno, se ve que esto es lo que se se mide cuando se homologa la cancha es la distancia entre bases. Después acá puede ser que haya un pequeño corrimiento. Igual lo que se ve acá es una combinación de de las dos cosas. Sí. De hecho, fíjense que el círculo este de pasto corta antes de lo que me marca de lo que me marca la proyección de de la cancha. Pero bueno, digamos, ningún algoritmo es 100% certero, por eso siempre, cada vez hay más y más soluciones disponibles. Eh, creo que visión por computadora es una de las áreas más activas en cuanto a a investigación. Eh, bueno, también tenemos modelos generativos donde por ahí la idea es alterar la imagen de alguna de alguna forma. Sí, esto se hizo con stable diffusion, pero digamos hay un montón de otras eh de otras opciones dando dando vuelta. Esto es algo bastante curioso. Bueno, nosotros usamos modelos de este tipo para resaltar detalles. Muchas veces las líneas de la cancha están un poco un poco borrosas, ¿sí?  
 

### 00:31:04

**lse posgrados:** o hay más eh más de una línea de tres dibujada en una cancha. Y lo que se hace o algo más curioso es que a veces hay tenemos esta cancha y por ahí después tenés un aro acá y otro aro acá. Entonces vos tenés otra cancha dibujada acá que es la cancha de los de los nenes que juegan que juegan así. Entonces claro, aquí es un quilombo detectar las líneas. Entonces, bueno, usamos modelos generativos que entienden un poco del contexto y pueden resaltarte las líneas que van, pero creo que acá el prompt era algo así como resaltame las líneas de la cancha en rojo, preservame la perspectiva. Y nos devolvió nos devolvió este frame, así que parece sacado de de una peli de de terror. Hay también ciertas ciertas herramientas, sí, que uno normalmente las usa en procesamiento de imágenes, pero que se pueden extrapolar a otra a otro tipo de datos. Eh, por ejemplo, uno puede trabajar con series temporales, ¿sí?, eh en otro espacio que, por ejemplo, puede ser el espacio de un espectrograma. Acá pareciera que el espectrograma es como otra señal, pero esto no sé, justo se da por el por el tipo de de información. En general puede haber formas geométricas completamente arbitrarias, un espectrograma, pero lo que uno hace es transformar una una serie de tiempo en una imagen.  
 

### 00:32:34

**lse posgrados:** ¿Sí? Y una vez que tiene eso, aplica algoritmos de procesamiento de de imágenes, ¿sí? para ver si están correlacionadas, si qué información puedo extraer, eh, o mismo entrenar un algoritmo de clasificación de, no sé, algo de teoría, detección y estimación de señales, pero sobre imágenes, ¿sí?, en lugar de eh series temporales. Con lo cual no es que solamente las herramientas de procesamiento imágenes sirven para para cosas como las que vamos a ver en la materia, sino que se pueden extrapolar a un montón de otras de otras áreas. Esto, siguiendo, ¿no?, un poco con la línea de los modelos generativos. Eh, esto creo que es bastante reciente, dice BO2, eh, actualmente creo que ya está disponible BO3 en en el AI Studio de Google, que es un editor de video donde vos le escribís qué es lo que queres que suceda y eh te genera secuencias como si fueran reales, pero en realidad nunca nunca sucedieron. Sí, nosotros en el laur lo usamos bastante para generar datasets y simular eh entornos y así como tenemos modelos eh de lenguaje que son modelos mixtos, en realidad procesan distintas distintas cosas. Para generar videos también tenemos lo mismo, pero para descripción. Sí. Eh, esto se hizo con quen 2, un dosingos grande lo habrán escuchado hablar.  
 

### 00:34:04

**lse posgrados:** O sea, con uno de los más chicos. Sí. Eh, se le se le mostró el video anterior. Sí. Y la la idea era ver si se pueden usar estos modelos como si fueran un clasificador. Sí. Esto nosotros lo probamos con miles de videos, no es que nos fijamos con uno. Pero en particular yo antes de la clase le pasé el video el video anterior, que es uno que saqué de de uno de los datas que tenemos y es bastante curioso porque son modelos que tienen la capacidad suficiente como para describir una acción. de una manera muy muy precisa, pero eh no tiene la capacidad de, entre comillas, razonar si la acción que están presenciando es sospechosa o no. Porque si uno lee la descripción que que dio el modelo, está está bastante bien. Sí. O sea, había una persona en un en un pasillo de supermercado, agarró un objeto, se lo puso adentro de, bueno, dice bolsillo, pero fue en la capucha en realidad, pero pero está bien. Y después te dice, "Mira, no hay indicación de que esto sea una actividad sospechosa, porque faltó agregar el prom se le pidió al modelo que dada una serie de videos, entregue este Jason, sí, tiene una aquí de bueno, clasifícamelo en sospechoso, o no sospechoso y agrégame una descripción.  
 

### 00:35:27

**lse posgrados:** Lo de la descripción es importante porque esto es algo que suele pasar bastante seguido. El modelo es bueno, incluso nosotros hacemos fine tuning de estos modelos, no no es que los usamos así como vienen, pero incluso con el fine tuning incluido, eh, a veces le pifian y le pifian en la tarea de clasificar. La descripción siempre suele estar bien. Entonces uno termina teniendo como un ensemble de cosas que dicen, "Bueno, con esto creo los triggers para elegir el el clip que de la cámara se lo voy a mandar al modelo de de descripción." el modelo de descripción me tira esta descripción y después si dice true le creo. Si dice false lo analizo, lo analizo un poquito más y se lo paso a un modelo de lenguaje que es muy bueno clasificando texto, ¿sí? Eh, que me puede decir dado una descripción si es una descripción compatible con algo sospechoso o es una descripción de algo normal. Entonces, uno después termina armando con un vector ahí con con información de cada modelo y termina decidiendo, bueno, ¿acepto o no la solución final que está que está ahí como como resultado. También podemos agarrar y dada una imagen tratar de estimar a qué distancia están los objetos que veo en la imagen, ¿sí? Que es lo que se conoce como estimación de de profundidad.  
 

### 00:36:54

**lse posgrados:** Eh, matemáticamente el problema de estimar distancia está resuelto. Sí, si uno tiene dos cámaras exactamente iguales, eh puede resolver la estimación de profundidad con una fórmula analítica. El tema es, bueno, conseguir dos cámaras exactamente iguales y eh si uno puede conseguir cámaras parecidas eh esa estimación es válida para cierto rango y para ciertas condiciones también. Entonces, y bueno, necesito dos cámaras, pero uno puede elaborar con redes neuronales que te dan capaz no una medición de distancia, pero sí te pueden decir qué cosa está atrás y qué cosa está adelante. Y muchas veces para algoritmos de tracking o algoritmos de segmentación eso ya es suficiente. Sí. Si uno quiere medir, bueno, necesita dos cámaras o laburar con con otro tipo de de sensores. Oh, perdón, recién veo la pregunta ahí que hicieron en el chat. Si tienen alguna duda y tienen micrófono, yo prefiero que abran el mic y me y me pregunten porque presento en otra compu y no suelo no suelo mirar mucho el el chat, o sea, parece que sí lo tengo la pantalla literalmente acá atrás. Entonces, no le suelo dar bola. Eh, las técnicas para reconstruir imágenes y esto 721 lo pregunta.  
**Juan Pablo Gonzalez:** Claro, sí, las que tienen están pixeladas.  
 

### 00:38:21

**lse posgrados:** Eran las imágenes de la primer diapo, ¿no? Sí. Sí, hasta sí hasta cierto mundo se puede e o sea, por ahí si están muy pixeladas como esas sí capaz que capaz que no, pero  
**Juan Pablo Gonzalez:** Tengo unos videos muy pixelados y busqué herramientas para reconstruirlo manera, pero es como inventar píxeles donde no lo hay. Es curioso, ¿cierto?  
**lse posgrados:** podemos probar a ver eh y mira lo ponemos yo tengo la cuenta de laburo ahí de del Google AI Studio. Podemos mandarle una ahora en la práctica y vemos que que genera este. No, no, no vas a poder. Esas están recontradestruidas, así que no ahí no hay no hay chance, digamos. Eh, algo te van a generar, pero va a ser cualquier cosa. Eh, sí hay técnicas, digamos, para cómo sí también depende cuánto cuánto estás dispuesto a pagar, digamos.  
**Juan Pablo Gonzalez:** Sí, vas a barras herramientas, que pregas herramientas y sigas cualquier cosa por ahora.  
**lse posgrados:** Lo lo de Google está buenísimo, ¿eh? Es un poco caro, pero está está bueno, funciona funciona increíble. No lo usé para restaurar. Sé que si vos tenés algo en baja resolución y lo querés pasar a 4K, lo podés hacer y anda muy bien, pero no si tenés una imagen que está recontrapixelada como las que vimos como las que vimos en en la clase.  
 

### 00:39:44

**lse posgrados:** Eh, después cosas que hay que digamos, no sé, mi tele, por ejemplo, lo tiene, hay consolas también que lo que lo hacen, es eh escalado de de resolución, pero con redes neuronales, no con interpolación. O sea, vos tenés algo que se renderiza a, no sé, 1280 por 720 de forma nativa, digamos, o sea, lo hace con con los drivers de la GPU y después eso lo pasa por una red neuronal que está en el hardware y eh te lo pasa 4K. Eso, eso se hace y existe, digamos, y se ve se ve bastante bien. Si si haces mucho zoom igual ves cositas raras, pero en los videojuegos se ve se se ve se ve perfecto.  
**Juan Pablo Gonzalez:** Gracias por el dato.  
**lse posgrados:** Ahora, ahora probamos agarramos una de esas imágenes y la la pruebo pruebo a ver qué me hace. Eh, bueno, nada, acá un como un recap, ¿no?, de de lo que estuvimos viendo. Eh, hoy hoy en día los algoritmos de visión están eh prácticamente todos lados. Por ahí hay ciertos lugares como el robot este que lleva cosas, son más cotidianos en otros países que que en el nuestro, digamos.  
 

### 00:41:00

**lse posgrados:** Por eso es una realidad. De hecho, yo cuando fue el año pasado me crucé uno de estos, yo estaba en Los Ángeles, de hecho era muy gracioso porque dejamos el auto con un amigo ahí en un parquímetro, fuimos a tomar algo y estaba un bicho, no era de Amazon Prime, sino que era como hay dos empresas que están ahí, están estos y otros que son blanquitos. Eh, estaba como una esquina y estaba dando vuelta, o sea, iba, se daba la vuelta, trataba de cruzar por otra calle, se volvía a dar la vuelta, cruzaba por la otra calle y estaba como en loop. Eh, el estacionamiento en esa ese lugar de Los Ángeles es válido por 2 horas nada más. Entonces, entre que dimos un par de vueltas, fuimos a comer, ya pasó una hora y media y yo dije, "Che, voy a ponerle ahí apoyar la la tarjetita en el parquímetro porque ya se nos va el tiempo. Eh, voy a a renovar el parquímetro." El bicho seguía en el mismo lugar, había pasado una hora 45 y seguía tratando de de cruzar la calle ahí por por las esquinas. Así que nada, dato curioso. Nada, cualquiera que haya usado un Oculus, sí, sabe que esto es un mix.  
 

### 00:42:08

**lse posgrados:** igual no solo de algoritmos de visión, sino que hay un input de sensores que generan cierta interacción con con el medio, ¿si? Y eso te hace ver determinadas cosas. Y ya sin ir más lejos, digamos, esto no no no estoy hablando de de un de un Tesla o algo por el estilo. Creo que cualquier auto de los últimos 5 años o incluso más eh tiene algo de ayuda de de visión por por computadora, desde, no sé, que te marca las líneas y te prende una luz en rojo cuando hay un objeto atrás o eh no sé, una visión 360, o sea, mi auto tiene y no es un un Tesla, o sea, es algo bastante bastante común, digamos. Eh, ah, me faltó agregar una que es la lectura de de las señales de de tránsito, que esa está está bastante buena porque te tira ahí que te estás zarpando de velocidad porque lee las señales que está viendo en la calle. Y si bien estas son aplicaciones relativamente modernas de los últimos por ahí 15 años, eh estoy seguro que más de uno estaba familiarizado, ¿sí?, con algoritmos de de visión. incluso desde chico, por lo menos los que más o menos tienen mi edad. ¿Alguien sabe qué qué es esto?  
 

### 00:43:30

**lse posgrados:** Lo que está acá en en la imagen es un Pokédex.  
**Juan Pablo Gonzalez:** Oke.  
**lse posgrados:** Bien. ¿Y cómo cómo funcionaba el Pokéex?  
**Juan Pablo Gonzalez:** tenía un software de visión por computadora.  
**lse posgrados:** O sea, pero ¿qué había que hacer con el Pokex? ¿Para qué para qué servía? Claro, era un como un aparatito. Esto es, no sé, año 94 más o menos, ¿no? O sea, el tipo tenía eh este aparato que tenía una cámara atrás, la enfocaba a uno de esos bichos que estaban ahí por el bosque y te tiraba como todas las estadísticas, ¿no?, del bicho a partir de, no sé, de una foto de de lo que sea que hacía y te decía todas las todas las características que tenían los los Pokémon. Y en esa época parecía algo bastante bastante fantasioso. Sí, pero hoy en día uno lo ve y se lo cree, dice, "Sí, esto es algo es algo común." O sea, yo lo podría implementar con una Raspberry, una Rubín o una pantallita, una pantallita LCD, digamos. O sea, no es nada no es nada complicado, pero fíjense que los estaban bastante bastante adelantados.  
 

### 00:44:45

**lse posgrados:** Allá en el 95, 94 decían, "Che, mirá, podemos tener esto que tiene un algoritmo que puede detectar el bicho, ¿sí? Te puede tirar la todas las todas las estadísticas." Y uno no se ponía a pensar si era posible o no. Uno lo tomaba como algo bastante fantasioso de ciencia ficción y hoy en día es algo recontrabásico dentro de lo que son los algoritmos de de visión. hacer un clasificador que dada la clase ha una query, una base de datos y te tira la la info. Sí. Bueno, y ya para cerrar toda la parte introductoria, ¿sí? E esta es como una línea del tiempo de cosas que están relacionadas a visión por computadora. Vamos a ver que, bueno, obviamente hay cosas que sí son propiamente dichas de visión, como los modelos de lenguaje con visión, como los que vimos hace dos diapositivas, algoritmos de segmentación que solamente tienen sentido cuando tengo una imagen para segmentar, pero después hay cosas un poco más abstractas, o sea, podemos hablar de eh cadenas de Markov, ¿sí? o campos de Marcov, podemos hablar de filtro de Calman, podemos hablar de eh había un par de cosas por acá. Bueno, por ejemplo, un filtro de partículas, ¿sí?  
 

### 00:46:06

**lse posgrados:** Este que son cosas que eh se usan en procesamiento de imágenes. De hecho, filtro de Calman hoy está en prácticamente cualquier tracker, sí, que se use. Eh, hoy en día tiene un filtro de Calman. Eh, podemos hablar, por ejemplo, de eh bueno, flujo óptico cuando prácticamente no teníamos video digital. De hecho, el paper de flujo óptico eh tiene integrales, o sea, está los ese algoritmo está planteado para un campo continuo, igual es es un delirio, o sea, no se integraban sobre celulosa y sobre los los rollos de de cine, no sé, no sé cómo lo lo pensaron implementar en su momento, pero son todo cosas que incluso algunas de, no sé, 1970 que eh hoy en día se están se están usando. De hecho, hay muchos conceptos de de deep learning eh que nacieron en los 90 o incluso en los 2000, la parte de redes convolucionales, por ejemplo, y recién hace 10 años se pusieron se pudieron poner en práctica porque no había hardware que se banque hacer ese procesamiento en un tiempo en un tiempo acotado. Bueno, y acá hay como un par de un par de áreas que son los capítulos del Selinski, el libro ese que yo le dije que es como el uno para para visión.  
 

### 00:47:26

**lse posgrados:** con con una especie de de glosario de distintas distintas áreas. Eh, bueno, ¿qué qué quieren hacer? ¿Seguimos un toque más? ¿Hacemos 10 minutos de de intervalo y después seguimos? ¿Está están ustedes? Sí, vamos a hacer un intervalo después más o menos grande 15 minutos antes de pasar a la práctica. Eh, pero no sé qué qué quieren hacer. Cuando le responda uno ya ya es suficiente. Bueno, seguimos seguimos un rato un rato más y de última hacemos un intervalo antes del del intervalo del intervalo grande.  
**Daniela Putrino:** No.  
**lse posgrados:** Eh, es charlar un poco sobre cómo es que nosotros percibimos la a nuestro entorno. Sí, es algo que tenemos que entender por lo menos muy por arriba antes de pasar a modelar eso con un software. Si nosotros tenemos células si que responden a distintas longitudes de onda. se estandarizó que si la longitud de onda que yo estoy midiendo está entre 400 y 500 nanm y estoy viendo algo que es azul, eh si está entre 500 y 600 es verde y si está entre 500 y 700 es el rojo.  
 

### 00:48:51

**lse posgrados:** Y acá hay como un gris en el medio que bueno, depende depende de la persona puede llegar a cambiar un poquitín. Eh, en nuestros ojos tenemos dos tipos de células, ¿sí? muy a a grandes rasgos. Sí, tenemos lo que se denominan bastones, que son un montón, de hecho, hay como un orden de magnitud más que las otras células, ¿sí? Que son los conos, pero eh responden poco a uno de los colores. Sí, acá estos serían como los colores primarios, pero acá hay un tema porque los colores primarios en pintura no son los mismos que en física óptica. Sí, es como que hay hay una distinción ahí. Para nosotros los primarios son estos tres y todas las combinaciones las formamos a partir de azul, verde y rojo, que si buscan cuáles son los primarios en pintura, no son estos tres y son y son otros. Como los bastones responden poco al rojo, eh, y yo necesito tres canales para poder formar todos los colores eh de una imagen, resulta ser que, bueno, no me sirven para ver color, pero sí me sirven para poder percibir sombras y texturas y contornos, incluso cuando no hay luz. Sí, pues son tantos que si yo estoy a oscuras y hay un as de luz dando vuelta, la probabilidad de que ingrese a iris y le pegue un bastón es alta.  
 

### 00:50:19

**lse posgrados:** Sí, es más alta que le pegue un bastón que un cono, porque los conos, que son las otras células tenemos entre 6 y 7 millones a próximo. Pero bueno, como responden unen poco a uno de los colores, solamente nos sirven para percibir formas y texturas. Los que sí nos sirven para percibir colores son los conos. Sí, estos responden a cada una de las tres eh rangos de longitudes de onda. Eh, y bueno, tienen esta distribución que yo la nombro porque después cuando veamos algoritmos para pasar de color a escala de grises, uno de esos algoritmos tiene en cuenta esta esta distribución. La percepción que nosotros tenemos de la intensidad no es lineal, o sea, no es que si yo pudiera medir con algún aparato la intensidad de la luz, voy a recibir la misma respuesta en mi cerebro. Sí, es no lineal. De hecho, sucede lo mismo con con muchos muchos sentidos. El oído es es uno por es un poco más más siguiente para los que no sé les gusta el audio, saben que los potenciómetros o las perillas que controlan el volumen no son lineales, sino que son logarítmicas, porque tienen que emular de alguna manera la percepción de volumen que nosotros tenemos con nuestro con nuestro cerebro.  
 

### 00:51:35

**lse posgrados:** Sí. Algo similar sucede con la vista. Por eso también todas las cámaras tienen algo que es eh un algoritmo de corrección gama que lo que hace es aplicar una función gama que es parecida a un logaritmo. Sí, es casi casi como la inversa porque es exponencial ahí dando vuelta. Eh, para deslinearizar la información que está entregando el sensor. Sí. La única manera que yo puedo trabajar con un sensor de silicio, bueno, hoy en día es el silicio, pero un sensor hecho un material semiconductor es usándolo en un régimen lineal. Sí, porque si es no lineal veo cualquier cosa. Pero claro, nosotros como no percibimos las cosas de una forma lineal, después tengo que aplicar una función para desinealizar ese esos niveles de intensidad que me está entregando. Eh, y bueno, esta estas células o la información de cómo se tienen que distribuir estas células eh están codificadas en nuestro ADN y resulta ser que eh están codificadas en un cromosoma en particular. eh que tienen las mujeres y no los hombres. Sí. Con lo cual, la probabilidad de que un hombre tenga daltonismo es mayor. Creo que uno es xx y el otro es xi. Eh, el el que tiene las dos x es el que el que lo es el que lo tiene codificado.  
 

### 00:53:06

**lse posgrados:** Eh, por ende, la probabilidad de tener daltonismo nombres es mayor porque, claro, tenemos como menos de esa información codificada en nuestro ADN para generar los conos, que son los responsables de que nosotros podamos percibir el el color. Bueno, ahora yo dije que el rojo, el verde y el azul como que se estandarizó, ¿no?, con esta con estas longitudes de onda, pero no dije cómo. La cosa es que en 1930 fue que se estandarizó esto en la comisión de CL y lo que hicieron fue poner gente a mirar una pantalla y a girar una perilla. Con la perilla modificaban la longitud de onda de la luz que estaba incibiendo iban anotando. O sea, le preguntaban, "Che, estamos en este en esta longitud de onda, ¿qué qué color estás viendo?" Y estaba todo bien, pero más o menos por acá había gente que decía, "No, che, yo estoy viendo yo estoy viendo el rojo." Y me tenía ni sentido porque me decían, "Che, no, pero todavía no llegaste al rojo, estás en la longitud de onda de del azul o del verde." Y no, pero yo veo el rojo, ¿eh? Y acá esto es curioso porque la longitud de onda es una magnitud medible, ¿sí?  
 

### 00:54:20

**lse posgrados:** Ahí no hay mucho margen para discutirlo, pero la percepción del color es un poco subjetiva, pero bueno, le pasado a mucha gente. Y acá surge el concepto de los colores metá metámeros, que son colores que tienen eh misma longitud de onda o están en el mismo rango de longitudes de onda, pero se perciben distinto. Eh, de alguna manera había que corregir esto. principio se modeló como una especie de distribución de valor negativo acá en el rojo solapada con el verde y con el con el azul, pero se llevó un modelo matemático, ¿sí? Que es un espacio de color distinto al espacio RGB. Sí, podemos pensar el RGB como un espacio cartesiano donde la terna RGB sí me determina un vector en el en el espacio, ¿eh? y estoy solamente en un cuadrante porque, bueno, todos los valores son positivos, pero digamos es un espacio clídeo con ciertas con ciertas restricciones. Eh, hay otros espacios, sí, el X y Z, por ejemplo, sí, eh, sí, sí, rojo, verde y azul.  
**Juan Pablo Gonzalez:** Perdón, me acabo de quería, me acabo de dar cuenta que RGB Red Green Blue destacar  
**lse posgrados:** Exactamente. De hecho, nosotros vamos a usar un montón de espacios de color que si bien los podemos podemos ver qué es lo que representan, nunca los vamos a poder representar, nunca nunca los vamos a poder visualizar porque todos los dispositivos están solamente pensados para reproducir RGB.  
 

### 00:55:54

**lse posgrados:** Sí. O sea, nosotros tenemos tres LEDs ahí en el en el cada monitor que tiene esas esos colores justamente y a partir de ahí forma todo lo demás. Podemos agarrar este espacio de color que resuelve el problema del color negativo, que es como un cambio de de base, pero no tenemos forma de visualizarlo, ¿sí? O sea, simplemente es RGB con un cambio de de base. Acá tiro este comentario que por ahora no tiene mucho sentido, pero cuando veamos otros espacios de color más adelante sí. Eh, este espacio y muchos otros, pero no el RGB, permiten separar las componentes de luz de las componentes de color. Y nosotros estamos muy acostumbrados al RGB, por eso no podemos pensar en modificar el brillo sin modificar el color. Si yo modifico el brillo de algo que es rojo y aumento el brillo, ¿no? De algo rojo y lo voy a ver rosado. Sí. Y si le saco brillo y lo voy a empezar a ver por ahí un poco más marrón hasta llegar a negro. Pero existen espacios de color donde no se da ese problema. Yo puedo tocar por un lado cosas que me modifican el color y por otro lado cosas que me modifican el brillo. ¿Sí? Y eso lo puedo hacer en espacios como el Sielab o el HSB.  
 

### 00:57:17

**lse posgrados:** El HSB es bastante bastante curioso y está bueno porque tiene una componente que está descorrelacionada de las otras, pero es un espacio cónico donde tengo el Hue que es el matiz que va de 0 a 360. En OpenC va de 0 a 180, pero digamos es lo mismo. Puede ir de er a uno siempre y cuando se se respete la la periodicidad de de esto que determine el color. Sí, el determina el matiz del color en realidad. Después sí sería como movernos sobre esta circunferencia. Después tenemos la saturación, que podemos pensarlo como algo que determina la intensidad, pero no en términos de brillo. Sí, determina que tan fuerte es el color que se mueve sobre el radio acá de este de este plato. Y después tenemos lo que se llama el value o el valor, que esto sí lo podemos graficar y esto es una imagen en blanco y negro o una imagen en escala de grises. En realidad, sí. Cuando vemos una película que está en escala de grises o nosotros cuando le decimos blanco y negra son grises, eh estamos viendo el canal del valu. Sí. Eh, el hue y la saturación no se pueden representar. Yo se las puedo pasar a mat plot lip y que me tiren un plot, sí, pero lo que voy a ver es cualquier cosa.  
 

### 00:58:40

**lse posgrados:** Sí. O sea, este está entre 0 y 1, este está entre 0 y 360 y encima el valor más cercano a a 360 entre, no sé, el cer el 200 y el 360 es el cero porque tiene distancia uno, pues el 360 vuelve al cero, con lo cual es un espacio muy útil pero medio complicado, o sea, es complicado para hacer clustering, es complicado para graficar, sí, pero al estar descorrelacionado de los otros dos es muy bueno para extraer características. Y bueno, el cel parecido, sí, pero eh es un espacio esférico. Hay un montón por ahí después podemos pegarle ahí una repasada a todos los espacios que hay. Eh, acá están las fórmulas. En OpenCB tenemos funciones para hacer la conversión de forma automática. Eh, en general nos vamos a estar moviendo entre el RGB y el HSB y bueno, y el de escala escala de grises que es solamente el valo. Sí, Daniela.  
**Daniela Putrino:** Sí, tengo una consulta. Eh, falta, o sea, faltaría como una variable, ¿no?, que sería la en el HCB, eh, porque, o sea, entiendo con el Hue que vos te parás en, o sea, en algún ángulo, pero eso además correlaciona con un color o se corresponde con un color.  
 

### 01:00:03

**Daniela Putrino:** Eso lo sabes por fuera o no que no sé que 120 gr es sian, por ejemplo.  
**lse posgrados:** Sí. E, o sea, hay tablas que mapean RGB, que es lo que vos podés interpretar, con una terna de HSB.  
**Daniela Putrino:** Sí, claro, eso faltaría, ¿no? Como ese mapeo, además de tener h saturación y value, ¿no?  
**lse posgrados:** Eh, a ver, si depende, o sea, si lo necesitas está, es cuestión de buscarlo, pero en general lo que uno hace es dice, "Che, hay cosas que en RGB no las puedo hacer." Sí, como por ejemplo, modificar el brillo sin tocar el color. Entonces paso a HSB, modifico la componente de value, la aumento o la bajo y después vuelvo a RGB.  
**Daniela Putrino:** Mhm.  
**lse posgrados:** O por ejemplo, eh, decís, "Che, yo quiero quiero hacer un un sistema que haga tracking de un objeto y, bueno, necesito usar características eh de la imagen, digamos, donde está el objeto, quiero saber los colores, la textura y todo eso. Y puede ser que en RGB se complique un poco, porque algo que para nosotros es fácil de ver, eh, por ir por algún algoritmo, ¿no? Pero sí es fácil de ver en HSB.  
 

### 01:01:20

**lse posgrados:** Entonces, digamos, si vos necesitás hacer una conversión de colores, sí, eh, pero en general no no tenés por qué dado un color en HSB, saber a qué corresponde en RGB.  
**Daniela Putrino:** Claro. Está perfecto. Bien, gracias.  
**lse posgrados:** De hecho, por ejemplo, algo que se da es que el blanco y el negro tienen el mismo valor de H. Sí. Pero lo que cambia es es el value, por ejemplo. Sí. O sea, uno tiene brillo value máximo y el otro tiene value mínimo, digamos. Se se da eso también de que para un color eh en en para un para un valor en Hu puedes tener dos colores tranquilamente en RGB.  
**Daniela Putrino:** Está buenísimo.  
**lse posgrados:** no se da en todos los casos, pero bueno, en el caso del negro sí, por eso no como que no no es tan importante la tabla.  
**Daniela Putrino:** Bien.  
**lse posgrados:** Eh, nosotros lo hemos usado en algún momento eh para hacer clustering, pero ahí teníamos como colores predefinidos y necesitamos hacer era más que nada para devolver porque el resultado se devolvía en en HSB y necesitábamos que el sistema lo lleve a la base de datos en RGB porque si no el front sabía qué hacer con eso. Pero no, no necesitas ninguna tabla. Bueno, eh, yo hagamos ahora sí 10 minutos, sí, hasta 8 y cuarto y y arrancamos con con lo que sigue y después sí vemos si metemos la práctica pegada o hacemos otro otro intervalo dependiendo del tiempo.  
 

### 01:02:57

**lse posgrados:** Bueno, volvemos 8 y cuarto entonces. Bueno, seguimos un ratito más. Tienen ahí un emoji o algo en alguna reacción en el chat, así sé que por lo menos hay una persona y arrangamos. Bueno, la idea es charlar un poco de todo lo lo previo que que uno hace antes de entrar a a escribir un algoritmo. Sí, porque el procesamiento de una imagen o un video empieza mucho antes de que esta imagen se digitaliza. Sí. Eh, hay cosas que eh nosotros no podríamos recuperar, ¿sí?, prácticamente con nada. Si la imagen, por ejemplo, se captura en un ambiente mal iluminado, ¿sí? O si tengo que monitorear un proceso y el hardware que estoy usando no es apropiado, ¿sí? Ya sea porque, no sé, el frame rate es muy bajo o porque el objeto pasa muy rápido y la lente no es lo suficientemente luminosa como para que yo pueda, no sé, setear un tiempo de exposición rápido y capturar la imagen. Sí. eh char hablar muy por encima de esas de esas cuestiones dentro de lo que es un un pipeline de procesamiento. Fíjense que lo que es el procesamiento en sí lo tenemos como casi al al final. Previo a eso, por ahí lo primero es decir, "Che, bueno, ¿qué cámara tengo que utilizar?"  
 

### 01:15:01

**lse posgrados:** Acá aparecen un montón de variables desde el tipo de sensor. Eh, el shutter es la manera en la cual yo leo la información de del sensor. Tenemos el tipo de interfaz, existen cámaras que eh de hecho yo tenía una, por ejemplo, esta de acá es una cámara con sensor de eh profundidad. O sea, esta no solo la podés usar para estimar profundidad, sino que tiene dos lentes para que vos puedas, no sé, hacer las ecuaciones y ver qué da. Pero tiene un sensor que te da eh una medición de profundidad para que puedas contrastarlo. Esto solamente tiene un interfaz USB. Entonces, si vos estás montando algo en una línea de producción, en una fábrica y capaz no tenés un cable USB cerca, ¿sí? y tengas que meter o un adaptador o por ahí algún dispositivo para hacer el procesamiento Edge y de ahí salís con con Ethernet o lo metés a una red sigby o salís por Wi-Fi o lo que sea. ¿Sí? Entonces, bueno, la interfaz es algo importante porque hay cámaras que solamente tienen una interfaz y no tienen otra. Eh, bueno, las loots son lookup tables, son operaciones matemáticas elementales que se hacen mapeando valores de una tabla a otra. Sí. Eh, parece medio básico, pero hay operaciones trigonométricas que se pueden hacer con esto, o sea, es bastante bastante complejo en realidad.  
 

### 01:16:25

**lse posgrados:** Eh, después el tipo de cámara, hay cámaras que uno le puede cargar algoritmos, hay cámaras, bueno, como la que les mostré yo, que hace una un cálculo y te dice la la distancia del objeto y hay un montón de de variables que seguro se me se me estén escapando. Después tenemos la lente, ¿sí? que tenés la lo que son las las telecéntricas, que son un caso muy particular y de estas estoy seguro que nunca vieron una. Ahora vemos un ejemplo. Y después las endocéntricas, que son las lentes comunes que se dividen todo en un abanico de lentes en función del campo de de visión que ofrecen. Incluso mismo dentro del campo de visión que ofrecen, no es lo mismo eh no es lo mismo la apertura que te da esa lente. Sí, hay lentes que son más luminosas que otras y por ende, bueno, más caras también. dentro de lo que es la iluminación. Esto básicamente es con qué luz y con qué ángulo voy a iluminar la la escena. Y después sí entra en juego el procesamiento y acá sí muy a grandes rasgos. Sí, yo lo dividí en dos porque yo puedo procesar en la nube mandando los datos a a algún lado o hacerlos en en situ.  
 

### 01:17:40

**lse posgrados:** Sí. O sea, tengo un dispositivo que hace el procesamiento en el mismo lugar donde está la cámara, digamos, lo dividí en dos porque cada una de estas de estas cosas tiene como su propia arquitectura. No es lo mismo una arquitectura Edge que una arquitectura cloud. Sí. Eh, y después adentro de esto también hay 1000 opciones porque en cloud por ahí no es lo mismo usar Cloud Run que tener una instancia 247 andando. Y en Edge no es lo mismo, no sé, un SP32 que una Nvidia Jetson o una computadora con GPU metida en un gabinete al costado de la cámara. Sí, digamos, son todo opciones que uno tiene que analizar al momento de hacer un un producto. Bueno, acá hay ejemplos de iluminación. Sí, fíjense que en función de cómo cambia el ángulo de incidencia de la luz, puedo ver eh más o menos detalles. Sí, por ahí si la idea es contar los pines de cada de este objeto, de esta placa, usaría backlight. Si quiero capturar detalles, usaría dark field, que es como luz rasante. Sí, la luz directa me me genera saturación, ¿sí? Con lo cual esto no lo puedo recuperar. Y bueno, luz con difusor es luz directa, pero eh se coloca como un domo así, no sé, algo que pareciera papel aluminio, una superficie muy refractante que hace que eh la luz cuando sale de la cámara lo que haga es incidir con cierto ángulo aleatorio, ¿sí?, en el objeto.  
 

### 01:19:08

**lse posgrados:** Y lo mismo cuando se refleja el objeto, la luz por ahí no va directo a la cámara y se se refracta y parte de o se refleja. Ya la verdad que no me acuerdo porque uno atraviesa y el otro y el otro no, el otro rebota. Igual siempre hay algo que atraviesa y algo que rebota. Es como sean los dos fenómenos. Eh, entonces bueno, la luz pierde parte de su intensidad y cuando ingresa la cámara ingresa con una intensidad menor y eh no hay saturación. Sí. Eh, no hay no hay mucho más en realidad para comentar acá. Eh, sí podemos hablar de que yo puedo iluminar con luz por encima o por debajo del espectro visible. Eh, esto me da varias ventajas. La primera es que yo puedo tener una cámara de vigilancia que ilumina con V o con ultravioleta y que capaz yo de noche a mí eso no me jode porque yo no puedo ver esa luz, pero sí el sensor de la cámara. Entonces yo puedo tener una cámara en una habitación completamente oscuras, pero que se ve todo como si fuera de día, porque la cámara tiene leds V o IR y está iluminando. Otras propiedades que tiene la luz por encima y por debajo del espectro visible es que puede atravesar eh la superficie del objeto y reflejarse en eh capas intermedias, ¿sí?  
 

### 01:20:25

**lse posgrados:** que me permite extraer cierta cierta información adicional de los objetos, ¿no? Acá tenemos ejemplos de billetes eh iluminados con con infrarrojo. Hay cosas que desaparecen a diferencia de del billete con una cámara normal, con una luz normal en realidad y eh bueno, hay cosas extra que puedan empezar a a aparecer también respecto de del lende. Vamos con el ejemplo de la telecéntrica. De la otra la verdad que no hay mucho para comentar. Eh, la telecéntrica lo que hace es que solo deja pasar. Esta es la lente, ¿sí? Este es el el eje óptico, sería ese centro de la lente. Eh, solamente deja pasar ases de luz que son paralelos al eje óptico. ¿Sí? O sea, si tienen un as de luz que está ingresando con un ángulo, dice, "No, este lo cancela." Eso hace que la imagen que se captura eh no tenga errores de perspectiva, ¿no? Acá el ejemplo creo que lo dice lo dice mucho mejor. Esto se usa cuando tiene uno tiene que hacer metrología, cuando yo tengo que medir, porque el error de perspectiva eh bueno, agrega cierta incerteza, ¿no?  
 

### 01:21:40

**lse posgrados:** De decir, "Che, ¿cuál es el área de del de la cabecita del pin?" Sí, es todo esto. Es todo esto. Sí. cambio con una lente telecéntrica, ¿no? Te ahorras un montón de dolores de cabeza, no hay error de perspectiva y se ve se ve perfecto. Después, bueno, la endocéntrica es la lente que todos conocemos. Sí, la clasificación a grandes rasgos es por ángulo, pero adentro de cada uno de de estas lentes se pueden eh se pueden subclasificar por lo que se conoce como el F#ARP. ¿Si? Esto lo que te dice es qué tan luminosa es la lente. Sí. Eh, con lo cual vos puedes tener en general cuando una lente es eh más cerrada en ángulo, se vuelve menos luminosa, pero hay lentes tele con un fsarp bajísimo. Está en la lente esa capaz vale $,000, tres veces más que lo que saldría la cámara, pero te da la misma luminosidad que te da un ojo de peso, un gran angular, ¿sí? Que tiene un montón de vidrio, es mucho más grande y deja pasar mucha más luz. Algo que estaba también dentro de ese pipeline era el sensor.  
 

### 01:23:00

**lse posgrados:** A grandes rasgos hay dos tipos. Sí, tenemos sensores de CCD que están quedando en desuso. Se siguen usando para aplicaciones muy específicas. Eh, por ejemplo, en astrofotografía se suelen usar CCD porque tienen mucha mayor eh sensibilidad y además uno no necesita la misma resolución que uno necesitaría en fotografía profesional, digamos, que no sea del espacio. Sí. Y después están los sensores COS, que hoy en día todo es COS o prácticamente todo. Tienen sus ventajas y su desventajas. Por ahí como ventaja del CCD. Supongo que yo tengo píxeles del mismo tamaño en silicio, tanto en Cemos como en CCD. Eh, bueno, todo el píxel es un elemento fotosensible. todo el área del píxel. La información viaja de del píxel hacia un conversor analógico digital que es único por sensor. En un SEMOS eh de mismo área de píxel parte del área es fotosensible, pero hay una parte que no, porque la conversión sí, de carga a nivel lógico se da en el fotositio, ¿sí? en el mismo en el mismo sensor. Entonces, yo lo único que hago acá es levantar la info de cada uno de los de los píxeles, con lo cual esto hace que la info de un sensor COS sea mucho menos ruidosa, porque yo digitalizo acá y lo que viajan son bits.  
 

### 01:24:31

**lse posgrados:** Sí, en un CCD la información es analógica hasta que llega al al conversor, lo cual, digamos, hace que sea mucho más propensa al ruido. Hay un par de cositas más que se pueden generar, ¿sí?, que es distorsión por blooming y smearing. Es cuando lo que viaja acá es carga, ¿sí? Este es un transporte por huecos o por electrones. Si la intensidad de luz es muy fuerte, esta carga se puede desbordar y puede pasar a los píxeles vecinos que capaz ya tenían carga que se va a desbordar a los vecinos y se termina desbordando. Y es lo que uno ve por ahí en las cámaras de vigilancia cuando entra un auto en escena y es de noche y uno ve como una pelota en los faros. y después la pelota desaparece y queda como una cruz en el medio hasta que desaparece también. Eso porque la carga se desbordó y se puede desbordar por filas o por columnas o en todo el sensor. En el SEMOS no existe ese problema, pero eh en función de cómo leo a la información puedo percibir la realidad de formas distintas. Eh, tenemos lo que es rolling shutter, que es leer por filas o por columnas, que puede provocar este tipo de distorsión.  
 

### 01:25:42

**lse posgrados:** O sea, tenemos un helicóptero que gira bastante rápido sus sus aspas y como yo estoy leyendo la información a un múltiplo de de la velocidad de giro, eh bueno, resulta ser que veo las aspas doblándose. Eh, esto no es un problema de tiempo de exposición, digamos. Si el tiempo de exposición estuviera mal seteado, simplemente vería las aspas difuminadas. Esto se salvó con un tiempo de exposición s super bajito, con lo cual eh la imagen es super nítida. Pero lo que veo es esta distorsión que es por por rolling shutter. Con Global Shutter yo leo todos los píxeles de un saque. Eh, obviamente para hacer esto necesito una cámara con un procesador mucho más potente. Es una cámara que suele ser más cara, eh, pero bueno, me da estos resultados. Veo las cosas más o menos como tienen que ser. Después hay hay más como clasificaciones porque dentro de tanto de CCD como SEMOS podemos hablar de cámaras reflex o cámaras eh mirrorless. Eh podemos hablar de eh cámaras que son full frame, cámaras que no son full frame. Sí, ya depende un poco más de de la geometría de la cámara más que de sensor. Sí. Eh, y acá dice microlentes, pero en realidad debería estar en los dos, porque por un lado sí yo de alguna manera tengo que compensar el hecho de que tengo menos área fotosensible, entonces uso microlentes que eh de  
 

### 01:27:13

**lse posgrados:** un de alguna manera eh redirigen toda la luz solamente al área fotosensible, con lo cual compenso un poco el hecho de tener menos área. Eh, pero también se usan microlentes en ambos casos. Porque hasta acá no hablamos de color. Sí. Eh, hay cámaras que tienen múltiples sensores. Sí. Un sensor para cada color, ¿sí? Por así decirlo. Un píxel para cada color en realidad. Eh, pero la mayoría lo que hace es poner algo que se denomina un arrey de ber, ¿sí? que es este arreglo de lentes donde cada una de las lentes filtra ciertas longitudes de onda. Entonces, yo tengo un conjunto de cuatro de cuatro píxeles, ¿sí?, que lo que hacen es tomar dos verdes, un azul y un rojo, y ahí yo devuelvo la información. ¿Sí? O sea, no es que tengo un sensor para cada color o algo por el estilo, simplemente hay lentes que dejan pasar cierta longitud de onda y omiten la otra. Entonces, yo después de ahí recupero el RGB eh de esa manera. Algo que también se usa y que no está acá, no está relacionado al color esto, sino al ruido.  
 

### 01:28:31

**lse posgrados:** Existen sensores que se denominan dual pixel. O sea, para cada para cada píxel que yo tengo en la imagen, hay dos píxeles de sensor encargados de tomar la información porque lo que hacen es toman de A2 y promedian. De esa manera bajan un poco el el piso de ruido que está entregando el sensor. Y bueno, esto es sería como el pipeline de una cámara, ¿sí? O sea, lo que tenemos la luz que ingresa la cámara, tenemos toda la parte de de de vidrio, de de óptica, ¿sí? Que es la lente. Acá lo que es la apertura es el control de apertura, ¿sí? Que básicamente es qué tan eh abierto dejo el iris de la cámara para que ingrese la luz. Tenemos el controlador de de shutter y después tenemos el sensor. Sí, seguido el sensor hay un módulo de ganancia. Es lo que en las cámaras se denomina ISO. Sí. Eh, no sé si alguien tiene una cámara profesional o semiprofesional. ¿Qué pasa si uno le sube mucho al al ISO? ¿Qué para qué sirve?  
**Jose Luis Diaz:** El era el grano viejo de la del de la película y lo que hace es tener más luz, ¿no?  
 

### 01:29:43

**Jose Luis Diaz:** Mientras más el más como que pareciera que capturaras más luz en principio.  
**lse posgrados:** Okay. Y y ¿qué pasa si te zarpas y le pones un montón?  
**Jose Luis Diaz:** Es que parece que tiene un grano más grande la la película.  
**lse posgrados:** Bien. Mm. Bueno, lo que hace el ISO es amplificar la señal que viene del sensor. Si yo saco una foto medio oscuras, eh, y digo, "Che, pará, está oscuro." Bueno, yo puedo amplificar esa señal. Entonces, los niveles de intensidad que eran bajitos van a subir y voy a ver la imagen como si fuera de día o la voy a ver más clara. Pero eh es un amplificador, es un control de ganancia. Y cuando yo amplifico una señal, ¿qué qué estoy amplificando además de la señal?  
**Juan Pablo Gonzalez:** ruid.  
**Ricardo Silvera:** Aleluya.  
**Daniela Putrino:** Salu!  
**lse posgrados:** Exactamente. O sea, uno cuando amplifica algo, amplifica la señal y el ruido. Después depende, ¿no? O sea, la relación señala a ruido de mi amplificador. Eh, va a ver qué tanto se amplifica la señal, qué tanto se amplifica el ruido, pero eh a priori uno amplifica un poquito de todo.  
 

### 01:30:47

**lse posgrados:** Eh, por eso es que si uno se zarpa de ISO, claro, también sube el piso de ruido de de la imagen y la ve la ve un poco más ruidosa. se ve un patrón de ruido, que es el que mencionó el compañero. Y bueno, ya acá a la salida del conversor, yo ya tengo la imagen, de hecho, eh bueno, en edición de fotografía uno suele elaborar con esto porque tiene un rango dinámico mucho más amplio, pero eh la mayoría seguimos de largo, ¿sí? Y tenemos eh un par de filtros. Este, el filtro de mosaico es para de alguna manera compensar los artefactos que nos genera el arrey de Bayer cuando filtra las longitudes de onda y nos devuelve la imagen. Sí, hay algoritmos después que sirven para enfocar la imagen y sacarle el ruido que tiene. Tenemos algoritmos de balance de blancos. Sí, muchas cámaras permiten setearlo para decir, "Bueno, lo llevo más para el lado de luz fría o luz cálida o o ha una compensación de color." Tenemos esto de la curva gama, que se los había comentado, ya es como un control eh de brillo y bueno, es el contraste en realidad, sí, un control no lineal de contraste que deslinealiza un poco la info que viene del sensor.  
 

### 01:32:07

**lse posgrados:** Después tenemos un algoritmo de compresión y ahí tenemos un PNG o un JP, ¿sí? o el formato que que sea. Y cuando estamos acá no tenemos rango dinámico amplio como en un row, porque en un row yo puedo estar con valores que van entre y uno, pero están en float y tengo, no sé, un millón de valores entre el cer y uno para cada píxel, con lo cual puedo puedo hacer un montón de cosas. En JPG o en cualquier otra imagen comprimida estamos laborando entre 0 y 25 por píxel. O sea, tenemos 256 valores posibles, con lo cual estamos acortando bastante el rango dinámico. Muchas veces esto está bien, eh, pero por ahí en otras aplicaciones vamos a querer estar acá. E bueno, y ya para cerrar la teórica y antes de pasar a la práctica, para nosotros una imagen va a ser una matriz que va a ser de n por m por 1. Si estoy en escala de grises o n por m, en realidad el por un lo tenemos cuando ya estamos laborando con tensores o n por m\* 3. Sí, para imágenes a a color. El rango ya lo lo mencioné recién. El cero representa el color más oscuro que puedo representar.  
 

### 01:33:30

**lse posgrados:** El 255 eh representa el blanco y esto entra en unign into, que es el tipo de dato que vamos a elaborar de forma nativa para las imágenes. Obviamente, si yo quiero hacer alguna operación matemática, eh probablemente voy a castear a float, voy a hacer la operación o las operaciones y después si necesito vuelvo a Ansign intoo laborando en en Flow. También vamos a ver que hay hay imágenes que si bien utilizan unign into, eh solamente van a tener dos valores, van a tener 0 y 255. Son imágenes binarias. Y eh en Open CB la imagen es una matriz, con lo cual se indexa como una como una matriz. Sí, vamos a trabajar por número de fila, número de columna y bueno, el número de canal. si estamos, no sé, en RGB, por ejemplo. Y esto lo claro porque cuando nosotros hablamos, por ejemplo, de la resolución de una imagen, lo decimos al revés, ¿sí? O sea, yo digo que mi monitor tiene una resolución de eh 1280 por, no sé, 1024, estamos diciendo que tiene 1280 en x, 1024 en i. Pero cuando hablamos del tamaño de una matriz, lo primero que decimos es la cantidad de filas, ¿sí?  
 

### 01:34:56

**lse posgrados:** O sea, decimos que tiene 1024 por 1080. Entonces ahí hay un tema como de nomenclatura, ¿sí? Cuando levantamos una imagen que no sé, está en 1920 por 1080, en realidad va a estar en eh 19, va a estar en 1080 por 1920. Tenemos un ejemplito. Sí. Eh, vemos como un gradiente que va de un de cer a 255 pasando por todos los valores intermedios y vemos que eh claro, para valores chicos partimos de de un negro absoluto y se va haciendo más claro a medida que se acerca al 255. Bueno, ¿alguna alguna consulta de de esto? Bueno, yo diría ahora sí metemos un intervalo de 15 minutos y seguimos con con la práctica que son y 38 y 39. Eh, bueno, volvemos 9 men 5 si les parece. Sí. Bueno, seguimos, seguimos un ratito más. Eh, bueno, esto lo van a ver en el en el repositorio, si tienen acceso al repositorio de de la maestría o de la especialización, que supongo que sí, eh, está en visión por computadora uno, clase uno, pero bueno, después de la clase o mañana les mando les mando el link igual.  
 

### 01:54:19

**lse posgrados:** Eh, vamos a ver un par de notebooks. Sí, capaz alguna se la dejo a a ustedes. Sí, leías ver por ahí la primera, eh, la segunda y por ahí vemos las las dos últimas que ya son un poquito más serias porque la idea es por ahí terminar la clase con eh Claro, sí, el link es creo que sí. Eh, le es terminar la clase con eh algún algoritmo, aunque sea básico, de segmentación de de objetos. Acá vamos a usar Numpie, vamos a usar OpenC y vamos a usar Matlot Lip solamente para graficar. Ya en otras notebooks. Después vamos a usar otras bibliotecas eh un poco más cómodas que nos ahorran varias líneas de de código. Para leer una imagen con OpenC tenemos el método de Yam Reid, que lo puedo usar así. Le paso el pad a la imagen y me va a levantar la imagen en el espacio de color predeterminado. Yo acá voy a agarrar y la voy a levantar en escala de de grises como para arrancar. Entonces le puedo pasar este flag que es y read. Eh, la imagen es una matriz de de bye. Sí, puedo pedir el shape, la puedo imprimir.  
 

### 01:55:42

**lse posgrados:** Sí, no hay no hay nada nada nuevo. De hecho, podemos hacer un print o Windows, se desconfigura todo el teclado. A ver, type de la imagen. Sí. Y vemos que es un arrey de de Numai. Para mostrar la imagen. La puedo hacer con con Matlot Lip. Tenemos la función IM show, ¿sí? Que me da esto, lo cual es un poco raro porque yo dije, "Che, pero la imagen está en escala de grises." Y efectivamente lo está. Yo no tengo tres canales acá. Bueno, resulta ser que Matlot Leip eh cuando recibe algo de forma matricial con YM Show dice, "Bueno, yo te lo voy a representar siempre por default como algo RGB." Sí. Eh, inventa los otros canales y te lo visualiza como si fuera una imagen a color. Eh, podemos visualizar con OpenC. Esto en general en Júpiter no funciona muy bien. Depende mucho de la versión que tengan. Puede ser que crashee todo, todo el kernel. Sí. Eh, pero Open CB sabe cómo visualizar sus imágenes, por eso la estamos viendo en escala de grises.  
 

### 01:57:00

**lse posgrados:** Cuando la grafico con Open C. Lo que no puedo hacer es visualizarla adentro de de la misma notebook, como una una ventana en bebida. ¿Qué pasa si quiero usar Matlot Lip para visualizar? Si no quiero usar OpenC. Bueno, le tengo que pasar el color map a mi imagen, a mi a mi función. El color map básicamente es una función, a cada uno le pasa la de la función que define cómo va a mapear los puntos, ¿sí?, de la matriz a colores RGB. Vamos a ver que podemos tener imágenes en punto flotante que con Open CB ni siquiera podemos visualizarlas. te tira error la función de visualización, pero sí podemos visualizarlas con mat. Eh, si le pasamos el color map adecuado, podemos visualizar eh cualquier cosa. Y fíjense que al pasarle el color map de Grey dice, "Ah, okay, estoy recibiendo una matriz, voy a mapear sus valores como si fuera eh escala de grises." Y acá se ve se ve bastante bien. ¿Qué podemos hacer? Podemos agarrar y graficar un perfil. Sí, me voy a parar en la columna 350 y voy a graficar esta tira, ¿sí?, de números, esta columna como si fuera eh una una señal, ¿sí?, como si fuera una serie temporal.  
 

### 01:58:22

**lse posgrados:** Y ya viendo esto, tenemos información que podemos extraer de la imagen. Sí, porque yo sé que, bueno, todo lo que está con este nivel de intensidad en torno al 75 es la mesa. Sí. Si este bloque estuviera un poquito más horizontal, eh, contando los píxeles que tengo de acá hasta acá, podría decir cuánto mide. Sí, por lo menos cuánto mide en píxeles. Si supiera a qué altura está la cámara y el tamaño del sensor, podría decir cuánto mide en milímetros. Sí, simplemente trazando trazando un perfil de de lo que estoy de lo que estoy mirando. Eh, también podemos ver dónde empieza y dónde termina la mesita. Sí, porque acá tenemos una zona clara y lo podemos ver acá como sube el nivel de intensidad y acá tenemos una zona un poquito más oscura que la mesa y lo podemos lo podemos ver. ¿Sí? Entonces, simplemente trazando, graficando una fila o una columna, ya podemos extraer información de del entorno. Sí, simplemente analizando los niveles de de intensidad y teniendo una vista de de superior, teniendo la imagen. Eh, podemos editar las las imágenes. Sí, por ejemplo.  
**Ricardo Silvera:** Disculpa, ¿hay el gráfico ese línea de píxeles de la de abajo o la de arriba?  
 

### 01:59:33

**lse posgrados:** Sí, sí, sí, sí. O sea, yo acá me paré en la columna 350, ¿sí?  
**Ricardo Silvera:** Ah, una columna.  
**lse posgrados:** Y dije, "Dame, dame todas las filas." Todas las filas de la columna 350. Y lo que tenés acá es una rey, es una rey de números que representa los píxeles de la columna.  
**Ricardo Silvera:** Ah, bien,  
**lse posgrados:** Eh, bueno, acá lo que hice fue agarrar la misma columna y ponerle ceros. Por eso vemos acá una línea, una línea negra. Sí, lo que hicimos fue pintar todo eso con con ceros. y le dibujamos dibujamos una una raya. Bueno, hasta acá imágenes escala de grises, pero la imagen original sí eh que la imagen bloque uno es esta de acá. Es una imagen a color. Eh, vamos a a levantarla en su en su espacio de color original. ¿Sí? Entonces, otra vez se ve un mre. No le pasamos ningún flag. Vemos que el tamaño de la imagen 480 por 640 por 3 porque está a color y la graficamos con matotli.  
 

### 02:00:47

**lse posgrados:** Pero hay una cosa medio rara porque esta es la imagen original y esta es la imagen que me está graficando Matot. Hay algo raro, no se ve igual. Sí, resulta ser que Open C no laura en RGB, sino que labura en BGR. O sea, el canal verde es el mismo. Por eso la mesa, si bien es un verde distinto, es el único color que no se no se alteró tanto. Sí, fíjense que el bloque parece azul, el fondo también cambió, pero la mesa es más o menos verde porque el verde queda en el medio y solo se swapean los de la punta. Eh, tenemos esta función que se llama convert color. Es la que vamos a usar después para cambiar entre distintos espacios de color. Acá lo que vamos a hacer es cambiar de BGR a RGB y ahí sí podemos visualizar con Matlotli. Si visualizamos con OpenC, no tenemos que hacer el cambio. Open CB sabe cómo visualizar sus propias sus propias imágenes. Ahora, tenemos que usar un método para hacer el cambio de color. Sí, no. Eh, esto es más rápido que la operación de de Numpai, por ejemplo. Y eh porque claro, en un yo, a ver, puedo operar con matrices, puedo hacer swap de canales, por ahí no se entiende del todo bien porque, bueno, yo si veo esto me tengo que poner a buscar que eran los dos puntos y el menos un eh, pero es la misma operación, estoy haciendo un swap, solo que Open CB lo hace de una manera un poquito más más rápida que que en un P. Esto capaz para una  
 

### 02:02:22

**lse posgrados:** imagen no tiene no tiene sentido, pero si uno trabaja con baches de imágenes y hay más procesos ahí en el medio, hacer un swap de de canales termina termina aportando bastante al frame rate de procesamiento. También si lo hago así, tengo que tener cuidado porque le pifiaste un a un índice y suapeaste en una dimensión que no iba y por ahí despejaste la imagen. Tenemos funciones, por ejemplo, para desarmar la imagen en en los cada canal por separado. Esto porque muchas veces yo quiero procesar la información por separado de cada uno de los de los canales. Lo podemos visualizar. Sí, acá lo lo grafico con un color map de gray porque digamos no hay otra opción para graficar esto. Estoy graficando cada canal por separado y podemos ver que bueno tienen distintos niveles de intensidad en escala en escala de grises. Podemos hacer el merch. Yo acá agarré y dije, bueno, hacemos el merge, pero anulamos el verde y el azul. Sí, dejamos solamente el rojo. Y esto es lo que se ve. Tiene sentido, ¿no? O sea, yo anulo el verde y el azul. Todo lo que queda son los tonos de componente rojo que estaban presentes en en la imagen.  
 

### 02:03:38

**lse posgrados:** No es que agregué algo nuevo, simplemente anulé verde y azul. Y esto es lo que lo que quedó. Eh, acá hago lo mismo, pero eh multiplico por una fracción en lugar de anularlo. Fíjense que reduzco un 10% y un 80 el verde y el y el azul. Este es el resultado. Así como yo grafiqué los perfiles de la imagen, el perfil en la imagen en escalga grises, puedo hacer lo mismo en la imagen, la imagen a color para los tres canales, la misma columna, ¿no? La columna 350. Y acá podemos ver algo, ¿sí? que eh por nos va a acompañar a lo largo de toda la materia y es que si bien numéricamente son valor, cada canal tiene valores distintos, presenta una correlación donde sube un canal, sube el otro, donde baja un canal, baja el otro, por ahí en proporciones distintas, pero es algo que se da. También se pueden dar casos donde uno sube y otro baja, pero siempre con una correlación, una correlación inversa en ese caso. Por eso es que el RGB no es un espacio de color muy cómodo para trabajar. Lo que nos da un canal eh por ahí sintetiza la información de los otros dos.  
 

### 02:04:57

**lse posgrados:** Sí. Eh, por eso van a ver que hay un montón de algoritmos clásicos que usan imágenes en escala de grises. Porque si yo agarro un poquito de cada canal y lo lo promedio en un número, termino teniendo la misma información que me dan los tres canales, pero en uno, con lo cual tengo eh menos cálculos para para hacer y no tengo información redundante. De hecho, vamos a ver por ahí más para el final de la materia que tenemos que armar eh sistemas de ecuaciones por píxel para poder estimar el desplazamiento de un objeto. Y en lugar de usar una imagen RGB y armar tres ecuaciones por píxel, tenemos que armar nueve ecuaciones sobre el plano, ¿sí?, alrededor del píxel, porque si usamos una ecuación por píxel en cada canal, tenemos ecuaciones linealmente dependientes debido a esta correlación. Y tenemos sistemas que no se pueden que no se pueden resolver. ¿Sí? Entonces, si bien uno puede pensar que tener color aporta información, para algunas cosas seguro que sí. Hay muchos algoritmos donde al contrario te está te está perjudicando porque introduce información redundante. Y después tenemos el método de IM WRght para poder escribir una imagen en en disco. Uno acá le pone la extensión que quiera y Open CB se encarga de buscar cuál es el el codec que tiene que usar para dejarla en un JPG, un PNG, etcétera.  
 

### 02:06:25

**lse posgrados:** Sí, Daniela.  
**Daniela Putrino:** Sí, te iba a preguntar justo eso. El formato de entrada también puede ser cualquiera o tiene que ser mapa de bicho, algo así.  
**lse posgrados:** No puede ser cualquiera porque Open CB por detrás usa FFNB.  
**Daniela Putrino:** Ah.  
**lse posgrados:** Entonces eh agarro depende. Creo que la lo que es Python Open CB usa FFNBEC. Uno puede compilar Open CB desde cero y pedirle que use Gstreamer, por ejemplo, que es otro framework, pero sí se se da cuenta solo. De hecho, vamos a verlo más adelante, pero si vos querés, por ejemplo, abrir un video que tenés en la compu entrega la cámara web o abrir un video que viene por streaming de una cámara por RTCP o HTTP,  
**Daniela Putrino:** Okay.  
**lse posgrados:** usas la misma función. Sí. No, código no cambia.  
**Daniela Putrino:** Ah, perfecto.  
**lse posgrados:** Open CB lo recibe y se da cuenta cuál es el el cuáles son la la secuencia de operaciones que tiene que hacer para devolverte un fotograma, por más que la fuente sea distinta.  
**Daniela Putrino:** Mich Yeah.  
**lse posgrados:** Bueno, eh la siguiente notebook es simplemente un rejunte de un par de ejemplos, ¿sí? eh de cómo podemos hacer anotaciones en imágenes.  
 

### 02:07:41

**lse posgrados:** Yo acá puse un par de funciones, pero hay un montón. Sí, la idea es que lean la la documentación de Open CB. Podemos hacer líneas, por ejemplo, para hacer líneas le pasamos coordenadas XI del inicio, XI del final, el color como una terna BGR, el ancho de la línea y el tipo de línea. Sí, esto por ahí importa cuando las líneas van en diagonal, si son rectas, sea por filas o por columnas, se pinta la fila o la columna, pero si quiero hacer una línea diagonal, bueno, cambia la función de cómo asigno color en esos píxeles que van a estar en la diagonal. Sí, porque tengo que hacer algún tipo de interpolación en algún momento. Sí. Eh, bueno, podemos también dibujar rectángulos. Esto lo usamos cuando, no sé, tengo un modelo que eh devuelve coordenadas Xi y quiero dibujar el bounding box de ese modelo. Tenemos una función que es rectangle. Tenemos la función de putext que recibe las coordenadas del texto y lo lo grafica. Vemos acá el ejemplito. Eh, podemos ajustar el texto a una región. ¿Sí? Entonces, podemos agarrar, usar, por ejemplo, get text size.  
 

### 02:08:56

**lse posgrados:** Dado el los parámetros del texto, ¿sí? Le pasamos el string, le pasamos la fuente, le pasamos el tamaño y el grosor y nos va a devolver el rectángulo que ajusta a ese texto. Esto lo usaríamos para, por ejemplo, agregarle acá un fondo. ¿Sí? Entonces, podemos graficar eh esto. ¿Dónde hago el putext? Esto yo lo haría todo blanco. Sí, sí, se nota un poquito un poquito mejor. Sí. Entonces tenemos el texto con un rectángulo de de fondo. Bueno, acá hay más operaciones con imágenes. Sí, esta esta notebook sí se la dejo se lo dejo a ustedes. Es medio cortita. Igual a mí me gustaría centrarme en estas dos espacios de de color. La idea es dada esta imagen, ¿sí? Eh, separar la mesa. Vamos a hacer de cuenta que la mesa es el espacio de trabajo en el que van a haber piezas. Entonces, yo quiero ignorar todo el resto de del fondo. Sí. Y lo quiero hacer de forma automática, digamos, porque yo podría agarrar y recortar una región, jarcodear una región ahí con coordenadas y usar eso.  
 

### 02:10:20

**lse posgrados:** Pero no sé si la cámara se mueve, por ejemplo, ese approach no me no me serviría. Tengo que tener algo que sea automático. Y creo que lo único que vimos en la teórica en cuanto a a teoría de de procesamiento o algoritmos son espacios de colores, así que vamos a encarar esto con, bueno, espacios de de colores y por ahí algo de de probabilidad y estadística. Eh, tengo todas estas conversiones de espacios de color. Sí, son un montón. De hecho, hay un montón. Hay algunas que yo ni siquiera usé. Así como yo tengo un ber a BGR, tengo un BGR a ber, así que tengo duplicados casi porque tengo el que va de un color a otro y de otro color al color original. Podemos ver cuántos son en realidad. Y lo dividimos por dos. ¿Dónde tengo el el divisor? Acá está shift 7, mira. Bueno, o sea, tenemos eh 187 conversiones de espacios de de color. Sí, yo creo que acá habré usado 10 como mucho. Escala de gris es HCB Y V. Eh, bueno, el si elab ahí son cuatro y capaz hay algún otro para usar transparencias, pero no mucho más.  
 

### 02:11:51

**lse posgrados:** Pero bueno, esta es la imagen. Sí, veamos cómo cómo podemos hacer para para separarla. Eh, algo que yo podría hacer es eh mostrarla en H en HCB. Sí. Acá ya vemos cosas que por ahí en RGB no se ven. Por ahí en el perfil no queda no queda tan obvio, pero sí se va a ver acá. Si yo paso el mouse por la imagen, van a ver que acá me va a aparecer arriba la terna de valores HSB. tengo la coordenada Xi y abajo la terna de valores. Si paso el mouse por la mesa, que en RGB tiene textura distinta y distintos, incluso distintos tonos, ¿sí? Distintos tonos de verde, en HSB, el Hue, ¿sí? Que predomina en la mesa está entre 80 y 83, ¿sí? No se mueve, no se mueve de ahí. Los otros dos valores sí tienen cierta cierta variabilidad, pero el Hugo está ahí casi clavado en en 80, ¿sí? Lo cual si yo, no sé, pongo paso el mouse por otro por otro componente dentro de la imagen, vemos que no en el bloque el H vale 30. Y bueno, en cada en cada en cada objeto tiene un valor característico porque claro, el hue es lo que determina el color y no hay otro objeto que sea igual de verde que la mesa.  
 

### 02:13:23

**lse posgrados:** Por eso no voy a ver ese hue repetido en otro lugar, ¿sí? O sea, el Hue ese que está entre 80 y 83 solamente corresponde a la mesa. Entonces, podríamos agarrar, ¿sí? y hacer una una pequeña descripción estadística, decir, voy a tener una muestra de de la imagen de la mesa en HSB. Voy a calcular la media, que es lo que lo que hacemos acá. Sí, esta es la el la porción que yo tomé. Esta es la muestra. Vamos a Ah, no, hice el print acá. Y está bien. Estaba entre 80 y 83, me da 81,47. Sí. Y vamos a usar esta función in range que recibe dos rangos, ¿sí? un rango alto y un rango bajo. Vamos a usar prácticamente el rango completo de todo. Digo, bueno, la media -10 y la media + 10, aunque tranquilamente podría hacerlo un +s1 y y un -1 como para agarrar eh mucho menos eh desvío y pedirle que me devuelva a todos los píxeles que están en ese rango.  
 

### 02:14:35

**lse posgrados:** Sí. donde el resto uso rango completo. Esto es lo que obtenemos. Sí, de hecho, por ahí el +10 y el -10 es un poco exagerado. Hacemos +1 -1, lo corremos de vuelta. Y esto así como está parece un poco un poco ruidoso, ¿sí? Pero vamos a ver que tenemos algoritmos que dada una imagen así la pueden dejar perfecta. Sí agarran y se encargan de limpiar puntos blancos en fondo negro y puntos negros en fondo blanco. Ya con este resultado, yo diría, bueno, con esto podemos podemos laburar bastante, o sea, ya es un resultado positivo, pero si dejamos el que estaba antes, sí, creo que era +10 y -10, vamos a poner + 5 y -5 para ser menos restrictivos, tenemos una mesa segmentada, ¿sí? Este, simplemente basándonos en una muestra de lo que quiero segmentar y una descripción recontrabásica, ¿sí? De decir, bueno, la media más menos un pequeño un pequeño rango para tener un un margen. Bueno, y por ahí lo que falta es aplicar la máscara a a la imagen, ¿sí? para obtener acá la mesa separada de todo el resto.  
 

### 02:15:53

**lse posgrados:** Para eso tenemos la función bitbice ant que recibe recibe la imagen y recibe la máscara. Entonces dice, "Bueno, aplico la operación ANT entre mi máscara que tiene píxeles que valen cero, píxeles que valen 255 la imagen. Solamente sobrevive todo lo que vale 255, ¿sí? Que es justamente la los píxeles de la mesa. ¿Alguna pregunta de esto? O sea, el mismo approach se puede usar para casi cualquier cosa que sea monotónica en términos de color y que no se repita en la imagen porque si no se nos complica. Sí, Daniela, la media objeto.  
**Daniela Putrino:** Sí, perdón, estoy repregunta, pero está buenísimo. ¿Hay alguna manera de sacar tipo un unique del Hue como para saber la media de cada objeto? ¿Sería una cosa así?  
**lse posgrados:** Sí, o sea, pasa que digamos podés convertir la imagen en un set donde, o sea, el canal del Hue en un set eh va a estar sobresegmentada. Sí. Eh, podrías por ahí hacer un histograma del Hue y trabajarlo en Bans, ¿sí?  
**Daniela Putrino:** Ah, okay, claro.  
**lse posgrados:** Ahí un poco para acotar un poco la solución, porque vos vas a tener un montón de Hes posibles, eh, menos que combinaciones RGB, pero van a ser un montón igual.  
 

### 02:17:09

**Daniela Putrino:** Sí, sí. Está bien, pero me imagin para ver esto un montón de 30 o 30 más 5, un montón de 80 más menos 5 y tener una idea de esos valores medios por objeto.  
**lse posgrados:** O sea, como mínimo 180 es que es el rango dinámico de del en CB. En la teórica vimos Claro, en en la teórica vimos que el Hue iba de 0 a 360, pero como les dije, depende mucho del framework, podría ir de 0 a un, ¿sí?  
**Daniela Putrino:** lo que estaba pensando. Ah.  
**lse posgrados:** o de 0 a 14, siempre y cuando esté esa periodicidad. Sí. Eh, sí, que nada, el tema de de del rango y eso depende del framework que no PCB CER a 180 difiere un poco de de la teoría.  
**Daniela Putrino:** Bien.  
**lse posgrados:** Bueno, le voy a decir un poquito más allá, ¿sí? Porque esto me sirve, pero claro, es una técnica super elemental y sirve cuando, bueno, no tengo algo que tenga un hue parecido, ¿sí? imagen y cuando el color del objeto que quiero detectar es homogéneo. Sí. Eh, vamos a subir un poquito la puesta, ¿sí?  
 

### 02:18:15

**lse posgrados:** Y vamos a hacer un detector, est un segmentador en realidad, pero de eh de morrones. Esta es la imagen y la idea es detectar el morrón amarillo. Vemos que el morrón amarillo tiene regiones que son más amarillas que otras, con lo cual tomar una muestra única acá ya no me no me sirve. Eh, tiene partes que son un poquito verdosas, no llegan a ser tan verdes como el morrón verde o como el tallo, pero son un poquito verdosos. Entonces tendría que tomar acá y como mínimo tres o cuatro tres o cuatro regiones para poder mapearlo de esa forma. Eh, y también eso me serviría capaz para este morrón, si el morrón se gira un poco o hay otro morrón con otra iluminación y tendría que tomar otras regiones, ¿sí? Porque seguramente eh la luz va a cambiar un poco. Entonces, tengo que hacer algo un poquito más avanzado, pero no tenemos tantas herramientas todavía. Eh, vamos a ver qué qué podemos hacer. Vamos a seguir el approach este de tomar una muestra, ¿sí? Y hacer algún tipo de de descriptor. Sí, lo vamos a hacer en RGB esta vez porque sabemos que claro, el H va a cambiar, o sea, si yo tengo algo que es amarillo y después es verde, el H va a ser distinto, con lo cual acá por ahí tomar el Hue eh hasta me puede llegar a perjudicar.  
 

### 02:19:38

**lse posgrados:** Lo tomo en RGB. Sí, sabemos que el RGB, bueno, es rojo, verde y azul, o sea, que existen las el todas las componentes ahí metidas en mayor o menor medida. Lo que voy a hacer es dada esta muestra tomar un descriptor, pero esta vez va a ser un descriptor estadístico. Voy a tomar la media y voy a tomar el desvío. Sí, como la imagen es una matriz de tres de tres canales, tanto la media como el desvío están en tres dimensiones. ¿Sí? Tengo una medida para el verde, una para el azul, otra para el rojo y lo mismo con el desvío. Y lo que voy a hacer es eh decir, bueno, devuélveme todos los píxeles esta vez que están dentro de eh un spam de seis siglas, ¿sí? tres y tres. En teoría, yo con esto estaría capturando, si fuera una distribución gauciana, el 99,7% de del espacio muestral. Y esto estaría estaría bien. De hecho, de hecho, sería la mejor solución si la distribución de color fuera una distribución eh gauciana. Sí. Bueno, procedemos igual que en el caso anterior, obtenemos la la máscara, ¿sí?  
 

### 02:20:56

**lse posgrados:** Aplicamos la operación ANT y esto es lo que nos da. Tenemos esta máscara. Hay cosas que van a caer por fuera de la distribución como el tallo del morrón. Ese ya no lo vamos a poder segmentar con nada, por lo menos no con este enfoque de color. Eh, pero bueno, acá tuve una región que era la más clarita que la perdí y también perdí parte de esta región que tenía algo de verde. Sí, porque claro, ese verde está por fuera de los seis sigmas. Si yo empiezo a agregar sigmas acá, sí está bien. Voy a capaz voy a capturar todo el morrón, pero voy a empezar a tener eh cosas que bueno, no son parte del morrón amarillo. Sí, es como que empiezo a abarcar cada vez más el espacio de color y termino agarrando cosas que no quiero detectar. ¿Sí? Entonces, veamos qué qué más podemos hacer. O sea, sabemos que esto nos da un baseline inicial, ¿sí? Basándonos en la estadística que está bien, pero me estoy quedando corto y le puedo sacar un poco más de jugo. Veamos cuál es la distribución de de colores. Sí, vamos a darla primero en en RGB.  
 

### 02:22:07

**lse posgrados:** Sí, Daniela.  
**Daniela Putrino:** Perdón, te paso una pregunta anterior. Me perdí. ¿En qué parte le pasaste que estabas mirando el morrón amarillo?  
**lse posgrados:** Acá le recorté un pedacito justo con esas coordenadas jarcodeadas porque sé que son justo de de una parte del morrón.  
**Daniela Putrino:** Ah, perfecto. Ah, listo. Gracias. Okay.  
**lse posgrados:** Bueno, acá lo que hago, sí, primero es graficar la distribución de colores en RGB. Bueno, no tiene mucha pinta de ser gauciano ninguno de los de los canales, ¿sí? Con lo cual está bien, vos podés tomar la media, ¿sí? Y y tomar los desvíos o valores que están dentro del rango, pero vas a abarcar cualquier cosa. Sí, de hecho es una distribución no paramétrica. Pero veamos cómo cambia esto si lo graficamos. en HSB. Bueno, se ve un poquito más eh un poquito más prolijo. Sí. Eh, de hecho, hasta podríamos pensar que hay canales en HSB que tienen una distribución obviamente no es gausiana, pero mucho más clasterizable que que en RGB. Esto que está acá, sí, el canal uno es el Hue.  
 

### 02:23:37

**lse posgrados:** Y fíjense que el color amarillo se concentra en un determinado rango de del Hue. Sí, yo podría agarrar este rango y decir, "Che, ni siquiera voy a tomar un un descriptor estadístico. voy a agarrar a partir de este análisis porque aparte yo sé que este valor de Hue que yo saco acá va a valer para todos los morrones amarillos, por lo menos iluminados con una luz parecida o con una cámara con características parecidas, ¿sí? Que sean más o menos amarillos también. Y esto me va a servir para segmentar cualquier morrón amarillo que tenga una distribución similar a al a este morrón. Sí. con lo cual es posible, digamos, todos los morrones amarillos se parecen. Entonces anoto cuál es este cuál es este rango. Sí. Y lo hago la hago corta. Digo, bueno, che, todos los otros uso el rango casi completo y el hue me quedo con todo lo que va entre 14 y 24. Genero la máscara y la aplico a la imagen como venimos haciendo. Este es el resultado que se obtiene. ¿Qué qué cosas quedaron afuera? El tallo que es verde, está completamente por fuera de la distribución y los brillos, pues los brillos son píxeles saturados que valen 255, que también están por fuera de la distribución porque algo saturado se corresponde al blanco tiene un hue que está por fuera de del rango que yo elegí.  
 

### 02:25:07

**lse posgrados:** Sí, pero todo lo demás está bastante bien y ganamos por un lado más área y tenemos menos falsos positivos que con la segunda solución eh que aplicamos cuando abrimos un poco el signo. Sí. Eh, bueno, nada, esto era la clase. Ah, me quedó me quedó una cosa pendiente que no me acuerdo quién había preguntado si se podía restaurar la imagen. Bueno, lo probé hacer. Me sorprendieron los resultados. Sí. Eh, por un lado usé WHK, sí, que es parte acá de de la suite eh de Google Pro, le pasé la imagen. Esto recibe un prompt, ¿no? No sirve para generar imágenes basadas en otras imágenes, como una especie de transfer learning. Sí, puedes crear videos y todo a partir de de prompts que pueden ser imágenes también. Y si bien no la mejoró, se dio cuenta qué es lo que estamos viendo. Sí. O sea, se avivó de que esto es una persona. Yo lo que le dije, che, mejorame la calidad de la imagen, nada más. Se dio cuenta que es una persona, sí, eh, que tiene un reloj acá le hizo un reloj gris parecido a este. No sé si le pegó o fue casualidad.  
 

### 02:26:20

**lse posgrados:** Bueno, acá le falta el reloj, pero digamos en términos de distribución de colores eh, está bien. De hecho, esta le hizo medio medio borrosa, pero un poco mejor que que esta imagen. Y después se lo pasé a Gemini. Sí. Eh, que yo partí de flash eh de flow, perdón, que sirve para hacer videos y importé el modelo de BeMI y lo que hizo es generarme un video a partir de esta imagen que está bastante bien, ¿sí? O sea, conceptualmente toma elementos de acá y las convierte a video. O sea, le pegó a la a la la al ángulo, a la perspectiva de de la mesa, un poco a la distribución también de de colores de la persona. Esto está bueno porque acá inventó como una silla que pareciera que está acá. Sí. Así que bastante bastante bien. Supo supo interpretar qué es lo que se está se está observando. Esto sí me llamó la atención porque acá pareciera que hay una repisa que yo en 4 años, esta es la primera vez que veo que hay una repisa acá medio blurriada y la la recreó. Sí, sí, bastante bastante sorprendente. Bueno, esta fue fue toda la teórica, ¿sí? Yo igual me quedo, no sé si tienen consultas o algo. Sí, Ricardo.  
**Ricardo Silvera:** Eh, te hago una consulta. E, ¿qué tiene en particular el color verde del del famoso fondo verde que ya usaba el Chapulín en los 70 para  
**lse posgrados:** Ah, el croma no es un color que no se repite nada más. O sea, vos podrías agarrar cualquiera, puedes poner un naranja brillante, pues sabés que no vas a tener nada de naranja brillante en la ropa y usar eso de croma, pero eso se estandarizó el verde por algo en particular de de la época, pero la idea de ayer usar un color que no se repita en nada.  
**Ricardo Silvera:** Ah, bien,  
**lse posgrados:** Entonces, bueno, justo ese verde ese verde fluor fue una fue una solución. Bueno, si no hay si no hay consultas, eh nos vemos nos vemos el jueves que viene o en la materia de los viernes. Y si están yo ahí igual estoy como como alumno en esa la de, ¿cómo se llama? Elementos de aprendizaje, operaciones de de aprendizaje automático mañana.  
**Jose Luis Diaz:** này.  
**Daniela Putrino:** Ops. Sí.  
 

### La transcripción finalizó después de 02:29:14

*Esta transcripción editable se ha generado por ordenador y puede contener errores. Los usuarios también pueden cambiar el texto después de que se haya generado.*