CONTROL POR GESTOS HACIENDO USO DE VISION ARTIFICIAL PARA FACILITAR EL USO DE PROGRMAS

INTRODUCCIÓN

PROBLEMÁTICA

El manejo del periférico de control por excelencia “el mouse”, se ha expandido y establecido en la vida diaria de cada una de las personas, pero su uso sigue siendo (en muchos casos) el mismo de hace años e incluso décadas, esto debido a dos factores: costumbre y falta de innovación.

La tecnología ha avanzado y hoy en día se trabajan con programas de diseño complejos capaces de trabajar con entornos en 3 dimensiones, pero el mouse no está a la altura de la tarea siendo difícil hacer acciones como: Rotar, Girar, Mover, entre otros.

Después del uso continuo de la mayoría de periféricos se percibe un cansancio por el uso de los mismos e incluso puede generar problemas más complejos como esguinces, síndromes como el túnel carpiano y la tendinitis; problemas que no están presentes en los gestos naturales que usamos todos los días.

Adicionalmente, el uso de cualquier periférico físico ocupa un lugar en el espacio de trabajo, los cables suelen ser molestos, y una alternativa Wireless resulta costosa y tienden a tener problemas de tiempo de respuesta y en ambos casos es limitado a una pequeña área útil dada por: pad mause o rango de conexión.

En un enfoque más empresarial, los empleados especializados en diseño o modelado tienen un uso normalizado de tabletas graficas para el diseño que son más precisas en su área de trabajo, pero aun así presenta los mismos problemas mencionados anteriormente.

JUSTIFICACION

La mejor forma de expresar, manipular y manejar algo es con las manos, medio con el cual se tiene más experiencia. Por mencionar un ejemplo, cuando se tiene un objeto en la mano y se desea rotar dicho objeto, basta con tan solo girar la palma de la mano; en un computador lo más probable es que se deba usar la combinación de un comando y una acción para realizar dicha tarea.

Además de eso, el manejo de un dispositivo o aplicación a través de gestos genera más libertad de espacio y opciones porque el espacio útil del control se desplaza a todo el espacio que las manos pueden manejar, teniendo como única limitante el ángulo de la cámara y su capacidad periférica; en cualquier caso, esto será superior y más cómodo a comparación de la tecnología actual.

Este proyecto busca reconocer gestos y comandos a través de sensores o una cámara especializada al rastrear e identificar la mano del usuario y los movimientos que está haciendo y como esto interactúa con la aplicación deseada a controlar. Para esto emplea el mapeado que hace uso de puntos de referencia situados en las articulaciones de la palma de la mano, con esto, se calcula la distancia de dichos puntos y el ángulo que forman para así detectar que gesto (en base a la posición de los puntos de referencia), se está ejecutando.

Esto puede ser expandido a el manejo de múltiples objetos y características de dispositivos como televisores inteligentes, luces, etc. Siempre y cuando dichos dispositivos puedan ser usados como parte del internet de las cosas.

MARCO TEORICO

https://pdfs.semanticscholar.org/67f2/87bccbd813304386d40c4518d1033f6ad089.pdf

* **Mediapipe:**

un entorno de trabajo multiplataforma de código abierto para construir “pipelines” para procesar datos perceptuales de diferentes modalidades, como video y audio. Este enfoque proporciona un seguimiento de manos y dedos de alta fidelidad mediante el aprendizaje automático (ML) para inferir 21 puntos clave 3D de una mano desde un solo cuadro. Mientras que los enfoques de vanguardia actuales se basan principalmente en entornos de escritorio potentes para la inferencia, este método logra un rendimiento en tiempo real en un teléfono móvil e incluso se escala a varias manos. (Google AI, 2019)

* **Pipelines:**

En arquitecturas de computadoras es una técnica que implemente paralelismo a nivel de instrucciones dentro de un solo procesador. En programación se basa en filtros y consiste en ir transformando un flujo de datos en un proceso comprendido por varias fases secuenciales, siendo la entrada de cada una la salida de la anterior.

Esta arquitectura es muy común en el desarrollo de programas para el intérprete de comandos, ya que se pueden conectar comandos fácilmente con tuberías (pipe).

También es una arquitectura muy natural en el paradigma de programación funcional, ya que equivale a la composición de funciones matemáticas. (Wikipedia, 2019)

* **Tensorflow lite:**

TensorFlow Lite es un conjunto de herramientas para ayudar a los desarrolladores a ejecutar modelos TensorFlow en dispositivos móviles, integrados y de IoT. Permite la inferencia de aprendizaje automático en el dispositivo con baja latencia y un pequeño tamaño binario.

TensorFlow Lite consta de dos partes fundamentales:

El intérprete TensorFlow Lite, que ejecuta modelos especialmente optimizados en muchos tipos de hardware diferentes, incluidos teléfonos móviles, dispositivos Linux integrados y microcontroladores.

El convertidor TensorFlow Lite, que convierte los modelos TensorFlow en una forma eficiente para uso del intérprete, y puede introducir optimizaciones para mejorar el tamaño y el rendimiento binarios.

TensorFlow Lite está diseñado para facilitar el aprendizaje automático en dispositivos, "en el borde" de la red, en lugar de enviar datos de un servidor a otro. (TensorFlow)

* **Non-maximum suppression:**

Este algoritmo es usado para asegurarse que, en una detención de objeto, el objeto detectado se detecte una sola vez (Aggarwal, 2018); este algoritmo funciona especialmente bien en casos de auto oclusión con las dos manos; como es el caso de un apretón de manos.

Un aspecto que a menudo se pasa por alto en muchos sistemas de detección es la etapa de Non-maximum suppression (NMS), que se usa para arrastrar múltiples cuadros delimitadores de alta puntuación alrededor de una instancia de objeto hasta una sola detección. Típicamente, esta es una operación de postprocesamiento aplicada al conjunto de cuadros delimitadores producidos por el detector de objetos. Como tal, no es parte de la función de pérdida utilizada para entrenar el modelo y cualquier parámetro debe ajustarse a mano. (Li Wan, 2015)

* **Anchors y Focal Loss:**

Las palmas de las manos son modeladas usando cuadros delimitadores oconocidos como “anchors” en una terminología de Machine Learning las cuales son unos cuadrados que se posan sobre los objetos detectados de la forma (X, Y, Altura, Ancho) (Christiansen, 2018).

Luego se usa un extractor de funciones encoder-decoder usado para una mayor conciencia del contexto de la escena e incluso para objetos pequeños, similar a RetinaNet un algoritmo de Facebook que hace uso de la perdida focal, esta se enfoca en muestras “difíciles” lo que mejora la precisión de la predicción (Tsang, 2019). (Zitnick, 2011)

* **Frames / FrameRate:**

La velocidad de fotogramas (expresada en fotogramas por segundo o FPS) es la frecuencia (velocidad) a la que las imágenes consecutivas llamadas fotogramas aparecen en una pantalla. El término se aplica igualmente a las cámaras de cine y video, gráficos de computadora y sistemas de captura de movimiento. La velocidad de fotogramas también puede llamarse frecuencia de fotogramas y expresarse en hercios. (Wikipedia, 2019)

Esquema de modelo para el seguimiento y reconocimiento de gestos usado por google:

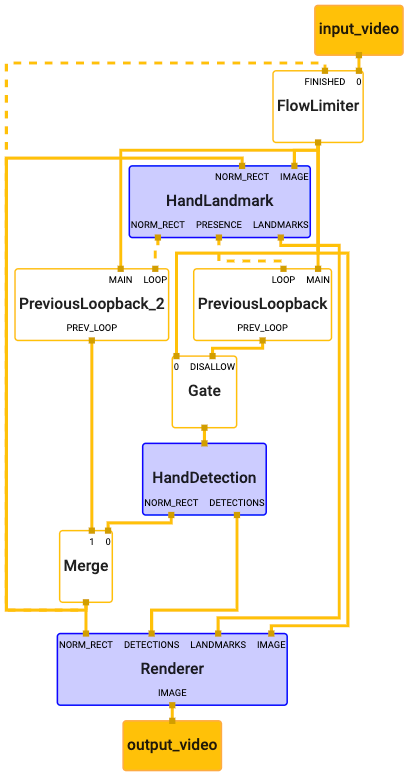


Figura 1. Mediapipe / GoogleAI

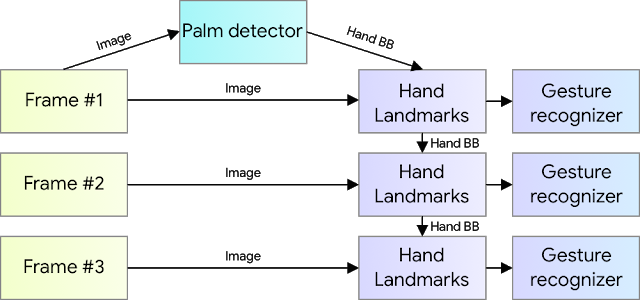


Figura 2

ESTADO DEL ARTE

* **Oculus Quest:**

La universidad de Standford creó un corazón virtual para ofrecerles a sus pacientes una visualización de los problemas que puede tener; la exploración de este corazón virtual se hace por medio de los controles de oculus rift el cual es un casco de realidad virtual; los controles que maneja emplean un acercamiento a los gestos humanos para controlar lo que sucede en la simulación, si bien no utiliza un completo uso de reconocimiento de gestos por medio de imágenes, es un punto medio entre estas dos tecnologías.

Sin embargo, desde entonces oculus ha mejorado su dispositivo de realidad virtual agregando el 25 de septiembre de 2019, reconocimiento de gestos por medio de inteligencia artificial en su totalidad por medio de cámaras monocromáticas lo cual va a reemplazar el uso de mandos o controles en dichos dispositivos de realidad virtual.



Figura 1 Oculus Quest

Se utilizan redes neuronales profundas para predecir la ubicación de las manos de una persona, así como los puntos de referencia, como las articulaciones de las manos. Estos puntos de referencia se utilizan para reconstruir una pose de 26 grados de libertad de las manos y los dedos de la persona. El resultado es un modelo 3D que incluye la configuración y la geometría de la superficie de la mano. (Facebook, 2019)

Se utiliza una nueva arquitectura de seguimiento que produce estimaciones precisas y de baja fluctuación de la posición de la mano de manera sólida en una amplia gama de entornos, y un marco de red neuronal cuantificado eficiente que permite el seguimiento de la mano en tiempo real en un procesador móvil, sin comprometer los recursos. (Facebook, 2019)

* **Kinect:**

The original relied on a structured lighting approach, which involved plastering a distinct pattern of infrared dots from its laser, somewhat like a QR code, on everything within its field of view. Knowing that objects farther away would have more distorted infrared dot patterns, the Kinect v1’sinfrared camera captured this distortion through triangulation and determined the depth of any scenery within its field of view. (Rahman, 2017)



Figura 2 Kinect V1

The Kinect can let us know whether a user’s hands are in any one of three recognized gestures or two non-gestures. The three recognized gestures, as shown in Figure, The Kinect for Windows v2 comes with Visual Gesture Builder, a tool that uses machine-learning techniques to help you train gestures and integrate them into your app. The tool is very easy to use and requires no prior knowledge of machine learning. (Rahman, 2017)

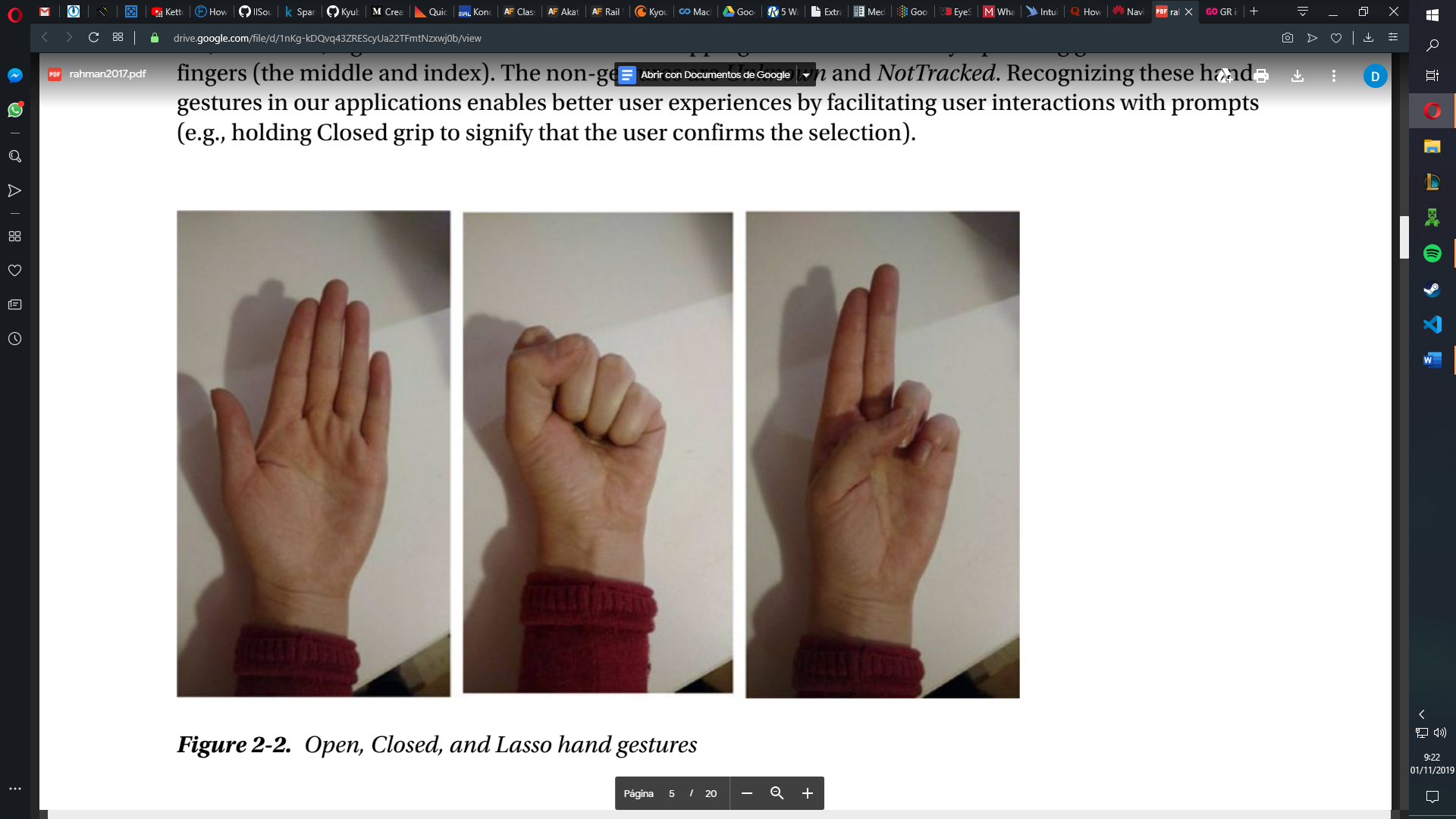


Figura 3 Gestos: Abierto, Cerrado y Laso

* **EyeSight:**

Esta tecnología basada en software (http://eyesight-tech.com/technology) utiliza procesamiento de imágenes en tiempo real y algoritmos de visión artificial para rastrear los gestos con las manos de los usuarios y convertirlos en comandos para un dispositivo. Touch Free procesa imágenes de una cámara de video 2D estándar, sensores infrarrojos o sensores 3D, y busca cualquier objeto que haya sido entrenado para detectar, como una mano, un dedo o la cara.

El sistema puede integrarse en casi cualquier tipo de dispositivo, rastrea e identifica movimientos y los convierte en comandos. (Garber, 2013)

* **Smartphone Touch control:**

Designed to support the larger screen experience whilst ensuring a simple, effective and optimal functional user experience. The new Gesture Navigation functionality makes it easier to acces more without having to lose a portion of your display with on-screen buttons or a larger bezel. (HUAWEI, s.f.)

<https://stfalcon.com/en/blog/post/intuitive-gestures-in-mobile-app-design>

* **Leapmotion:**

Desde una perspectiva de hardware, el Leap Motion Controller es bastante simple. Consta de dos cámaras y tres LED infrarrojos. Estos rastrean la luz infrarroja con una longitud de onda de 850 nanómetros, que está fuera del espectro de luz visible; los datos toman la forma de una imagen estéreo en escala de grises del espectro de luz infrarroja cercana, separados en las cámaras izquierda y derecha.

A pesar de las ideas erróneas comunes, el Leap Motion Controller no genera un mapa de profundidad, sino que aplica algoritmos avanzados a los datos sin procesar del sensor y luego procesa las imágenes. Después de compensar los objetos de fondo (como cabezas) y la iluminación ambiental del entorno, las imágenes se analizan para reconstruir una representación 3D de lo que ve el dispositivo.

A continuación, la capa de seguimiento coincide con los datos para extraer información de seguimiento, como dedos y herramientas. Los algoritmos de seguimiento interpretan los datos 3D e infieren las posiciones de los objetos ocultos. Se aplican técnicas de filtrado para garantizar una coherencia temporal uniforme de los datos. El Leap Motion Service luego alimenta los resultados, expresados ​​como una serie de cuadros o instantáneas, que contienen todos los datos de seguimiento, en un protocolo de transporte.

La biblioteca cliente organiza los datos en una estructura API orientada a objetos, gestiona el historial de cuadros y proporciona funciones y clases auxiliares. A partir de ahí, la lógica de la aplicación se vincula con la entrada Leap Motion, lo que permite una experiencia interactiva controlada por movimiento. (LeapMotion, 2014)

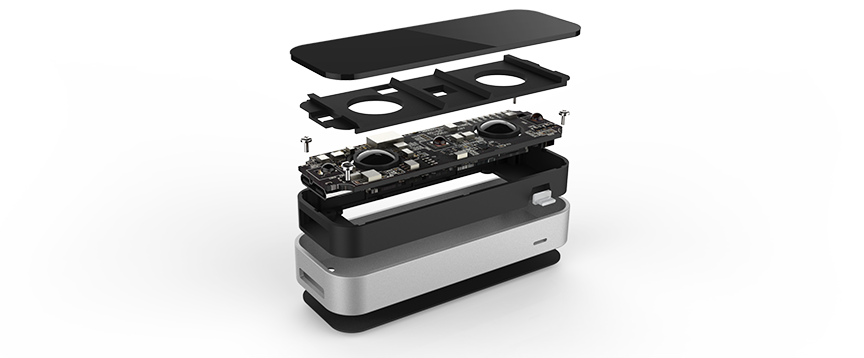


Figura 4 Leap Motion

* **Google Mediapipe:**

El acercamiento de GoogleAI a este problema es dado por el aplicativo conocido como Mediapipe el cual fue publicado el día 19 de agosto de 2019 el cual busca una solución en tiempo real para seguimiento y detección de manos. Este algoritmo utiliza técnicas de Deep learning, non-maximum suppression, perdida focal, entro otros. Este fue entrenado con 3 datasets siendo uno proveniente del mundo real e imágenes tomadas a las palamas de la mano que posteriormente fueron debidamente clasificadas, el mismo proceso se aplicó con un dataset hecho con palmas de manos renderizadas en un entorno 3D, y el ultimo dataset una combinación de los dos datasets anteriores (Google AI, 2019)

Precision de regression

|  |  |
| --- | --- |
| Dataset | Mean regression error  normalized by palm size |
| Only real-world | 16.1 % |
| Only rendered synthetic | 25.7 % |
| Mixed real-world + synthetic | 13.4 % |

# Referencias

Aggarwal, S. (30 de 08 de 2018). *Quora.* Obtenido de Quora: https://www.quora.com/How-does-non-maximum-suppression-work-in-object-detection

Christiansen, A. (15 de 10 de 2018). *Medium.* Obtenido de Medium: https://medium.com/@andersasac/anchor-boxes-the-key-to-quality-object-detection-ddf9d612d4f9

Facebook. (25 de 09 de 2019). *Facebook Artificial Inteligence.* Obtenido de Facebook Artificial Inteligence: https://ai.facebook.com/blog/hand-tracking-deep-neural-networks

Garber, L. (31 de 10 de 2013). *IEEE Xplore.* Obtenido de IEEE Xplore: https://ieeexplore.ieee.org/abstract/document/6649963

Google AI. (19 de 08 de 2019). *Google AI Blog.* Obtenido de Google AI Blog: https://ai.googleblog.com/2019/08/on-device-real-time-hand-tracking-with.html

HUAWEI. (s.f.). *Huawei Support*. Obtenido de https://consumer.huawei.com/uk/support/faq/navigate-your-phone-using-gesture-control/

LeapMotion. (09 de 08 de 2014). *Blog LeapMotion.* Obtenido de Blog LeapMotion: http://blog.leapmotion.com/hardware-to-software-how-does-the-leap-motion-controller-work/

Li Wan, D. E. (2015). End-to-End Integration of a Convolutional Network, Deformable Parts Model and Non-Maximum Suppression. *IEEE Xplore*, 851-852.

Rahman, M. (2017). *Beginning Microsoft Kinect for Windows SDK 2.0.* Apress. Obtenido de https://drive.google.com/file/d/1nKg-kDQvq43ZREScyUa22TFmtNzxwj0b/view

TensorFlow. (s.f.). *TensorFlow.* Obtenido de TensorFlow: https://www.tensorflow.org/lite/guide

Tsang, S.-H. (24 de 01 de 2019). *Towards Data Science.* Obtenido de Towards Data Science: https://towardsdatascience.com/review-retinanet-focal-loss-object-detection-38fba6afabe4

Wikipedia. (30 de 07 de 2019). *Wikipedia.* Obtenido de Wikipedia: https://es.wikipedia.org/wiki/Arquitectura\_en\_pipeline\_(informática)

Wikipedia. (03 de 10 de 2019). *Wikipedia.* Obtenido de Wikipedia: https://en.wikipedia.org/wiki/Frame\_rate

Zitnick, D. P. (2011). Human-debugging of machines. *In NIPS WCSSWC*.

<https://www.infotechnology.com/negocios/Desarrollan-un-sistema-de-reconocimiento-de-gestos-utilizando-senales-Wi-Fi-20130619-0001.html>

<https://core.ac.uk/download/pdf/153562887.pdf>

<http://catarina.udlap.mx/u_dl_a/tales/documentos/mcc/lopez_r_o/capitulo1.pdf>

<http://opac.pucv.cl/pucv_txt/txt-0500/UCC0990_01.pdf>

<https://www.europapress.es/portaltic/sector/noticia-google-desarrolla-herramienta-android-analiza-manos-entender-gestos-lengua-signos-20190822175222.html>

<https://www.stanfordchildrens.org/es/innovation/virtual-reality/stanford-virtual-heart>

<https://www.researchgate.net/publication/322263316_Design_and_Implementation_of_Gesture_Controlled_Robotic_Arm_for_Industrial_Applications>