# **Detección Automática de Distracciones Durante el Estudio Usando la Webcam**

**Visión a Futuro: Hacia un Pomodoro Personalizado**

## **1. Introducción y Planteo del Problema**

El aumento del estudio remoto y el uso intensivo de computadoras generó nuevas dificultades para los estudiantes a la hora de mantener la concentración. En casa, las distracciones son frecuentes: mirar el celular, alejarse de la pantalla o incluso quedarse dormido. Estos hábitos afectan el rendimiento académico y la productividad.

Este proyecto propone una solución basada en visión por computadora, que utiliza la webcam para detectar en tiempo real si el estudiante está atento, distraído o somnoliento, generando un reporte visual y un dataset con el tiempo pasado en cada estado. Así, el estudiante puede conocer sus hábitos reales y plantear estrategias para mejorar su rendimiento.

## **2. Objetivos**

**Objetivo general:** Desarrollar una herramienta que, utilizando la webcam, detecte y registre automáticamente los periodos de atención y distracción durante sesiones de estudio.

**Objetivos específicos:**

* Detectar si el estudiante está prestando atención a la pantalla o se distrae durante la sesión de estudio.
* Identificar eventos como mirar hacia otro lado, cerrar los ojos por mucho tiempo o quedarse dormido.
* Generar un informe visual y estadístico del tiempo en cada estado (atención, distracción, somnolencia).
* Guardar la información de cada sesión en un dataset estructurado, preparado para análisis posteriores.

## **3. Diseño de la Solución**

La solución se desarrolló en Python utilizando librerías de visión por computadora y procesamiento de imágenes:

* **OpenCV** para captura y procesamiento de video en tiempo real.
* **MediaPipe** para detección de rostro y landmarks faciales.
* **PIL y Matplotlib** para visualización y gráficos.
* **pyttsx3 y winsound** para alertas de voz y pitido.

**Pasos principales:**

1. Captura de video desde la webcam.
2. Detección del rostro y landmarks faciales usando MediaPipe.
3. Análisis de la dirección de la mirada y del estado de los ojos para clasificar el estado:  
   * “Atento”
   * “Distraído”
   * “Somnoliento”
4. Visualización en tiempo real del estado actual sobre la imagen.
5. Al finalizar la sesión, generación automática de un **dataset CSV** y gráficos resumen (porcentaje de tiempo en cada estado, cantidad de distracciones, etc.).

## **4. Implementación**

El desarrollo se realizó en scripts Python, para facilitar su prueba y visualización. Se utilizaron umbrales simples para diferenciar entre los estados. Además, el sistema es capaz de emitir un pitido y un mensaje de voz (“Volvé al estudio”) ante distracciones prolongadas, ayudando al usuario a retomar la concentración.

**Guardado de datos:** Al terminar la sesión, se genera automáticamente una fila en el archivo dataset\_distracciones.csv y un resumen en resumen\_sesiones.csv, listos para análisis futuros o visualización con scripts adicionales.

## **5. Resultados y Evaluación**

El sistema permite detectar en tiempo real cuándo el estudiante está atento, distraído o somnoliento. Al finalizar cada sesión se generan:

* Un gráfico de torta que muestra el porcentaje de tiempo en cada estado.
* Un resumen estadístico con cantidad y duración de distracciones.
* Un registro acumulativo (dataset) de todas las sesiones, permitiendo comparar entre días y analizar tendencias.

**Ventajas:**

* Brinda información clara y visual sobre los hábitos de estudio reales.
* Ayuda a tomar conciencia de las distracciones y mejorar la productividad.

**Limitaciones:**

* La precisión depende de la calidad de la cámara y de la luz ambiente.
* No distingue todos los tipos de distracción (ejemplo: usar el celular sin apartar la mirada).
* No implementa alertas automáticas para todas las situaciones (solo para distracciones prolongadas).

## **6. Conclusiones y Proyección Futura**

El proyecto demuestra que es posible, usando técnicas de PDI y visión por computadora, detectar en tiempo real el estado de atención de un estudiante. Esta herramienta es útil para cualquier persona que desee mejorar sus hábitos frente a la computadora.

**Visión a Futuro:** Todo el sistema y los datasets están pensados para, a mediano plazo, **desarrollar un modelo Pomodoro personalizado**. Con la información recolectada, será posible ajustar dinámicamente los intervalos de estudio y descanso según el comportamiento real del usuario, generando recomendaciones adaptadas a cada persona y optimizando así sus sesiones de estudio.

**Nota:** El análisis avanzado y la implementación de este Pomodoro adaptativo no forman parte de este trabajo, pero la arquitectura del sistema ya está diseñada para facilitar esa evolución en una siguiente etapa.

## **7. Posibles mejoras**

* Mejorar la robustez de la detección de distracción (usando más landmarks o modelos basados en aprendizaje profundo).
* Incluir detección de objetos (como celulares) para distinguir diferentes tipos de distracción.
* Implementar alertas más personalizadas o adaptativas según el historial de cada usuario.
* Adaptar el sistema para funcionar en distintas condiciones de luz y tipos de cámara.

## **8. Anexos**

* **Scripts de análisis y visualización de datos:** Se incluyen scripts (grafico\_resumen\_sesiones.py, grafico\_promedios.py, grafico\_concentracion.py) para visualizar la evolución de las sesiones, el promedio de concentración/distracción, y la cantidad de distracciones.
* **Entorno virtual:** Se provee un archivo requirements.txt y guía para replicar el entorno de desarrollo.