EDICIÓN #007

En 2026 el 90% de Internet será IA, ¿Cómo sabremos qué es real? | Google SynthID

12 de Enero de 2026

En 2026 el 90% de Internet será IA, ¿Cómo sabremos qué es real? | Google SynthID

Te explico SynthID, la tecnología de watermarking estadístico que marca texto, imagen y vídeo para evitar la teoría del internet muerto y el colapso de modelos.

12 November 2025

Contenido

El viernes pasado tuve la suerte de participar en el VallaTech Summit, un evento centrado en tecnología, IA y desarrollo organizado por el Google Developer Group Valladolid.

Fue una experiencia increíble. Disfruté mucho del evento y aprendí un montón de las ponencias y de la gente que conocí allí. Gracias a todo el equipo del GDG Valladolid por hacerlo posible y por darme la oportunidad de estar allí compartiendo lo que más me gusta.

ValláTechSummit 2025

Al evento llevé una charla sobre **uso irresponsable de la IA**: algunas de las maldades que ya se están haciendo hoy con modelos generativos y, sobre todo, cómo protegernos de ellas y proteger nuestro propio contenido frente a su uso indebido en entrenamientos y redes.

He dejado todo el material de la charla disponible para que podáis revisarlo con calma:

- Presentación sin anotaciones
- Presentación con anotaciones ← Este es el interesante si quieres profundizar o probar los ejemplos por tu cuenta.

Y precisamente una de las partes que más me gustaron, y de la que hoy quiero hablar más a fondo, fue **SynthID**, la tecnología de Google para **identificar contenido generado por IA** y que creo que deberías conocer porque cada vez va a tomar más importancia.

¿Qué es SynthID? Guía práctica sobre la marca de agua para contenido generado por IA

SynthID es una tecnología desarrollada por **Google DeepMind** para marcar de forma invisible los contenidos generados o modificados por modelos de IA: texto, imagen, audio y vídeo.

Su propósito es sencillo: hacer visible lo invisible. Permite a empresas y usuarios identificar **cuándo un contenido ha sido generado o alterado por IA**, fomentando la transparencia y la confianza en el uso de herramientas generativas.

Google la está integrando directamente en casi todos sus productos de IA como Gemini para que cada vez que se genere un texto, una imagen o un vídeo, se inserte una marca digital imperceptible para los humanos pero detectable por sus sistemas.

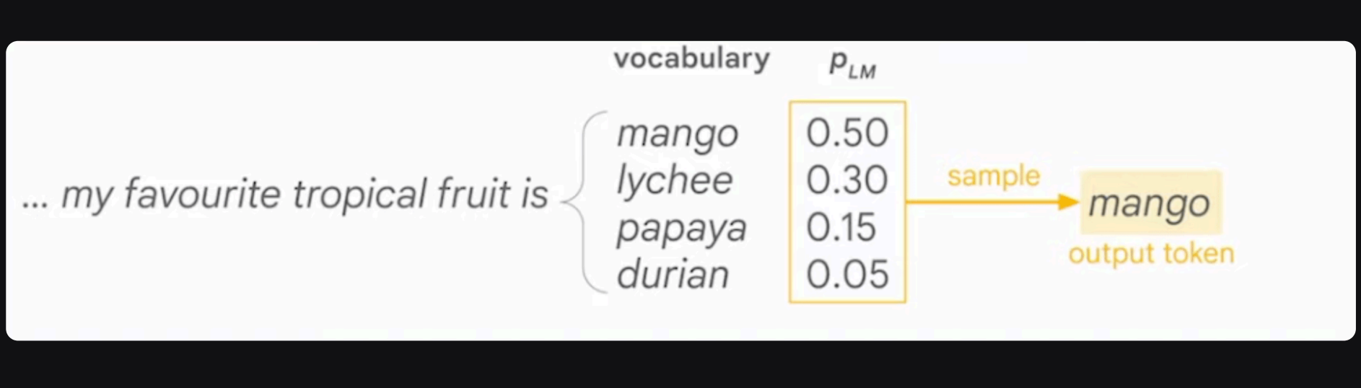
¿Por qué lo hacen? Según las estimaciones actuales, más del 90 % del contenido en Internet será generado por IA antes de 2026. Si las compañías no pueden distinguir lo humano de lo sintético, **acabarán entrenando nuevos modelos sobre material ya generado por IA**, creando lo que comentaba en la charla como el problema de las fotocopias de fotocopias: modelos entrenados sobre datos sin valor nuevo donde las ideas humanas se van perdiendo y donde la **teoría del internet muerto** va cobrando más sentido.

SynthID busca precisamente evitar eso, aportando un mecanismo estándar que permita preservar la trazabilidad y el origen del contenido digital, manteniendo la calidad y la originalidad en la era de la IA generativa.

Cómo funciona SynthID en texto: Watermarking estadístico para detectar texto generado por IA

Cuando un modelo de lenguaje (LLM) genera texto, el proceso habitual comienza calculando una **distribución de probabilidad** sobre los posibles tokens siguientes.

Por ejemplo, tras una frase como "*my favourite tropical fruit is*", el modelo podría asignar mayor probabilidad a "*mango*", "*lychee*" o "*papaya*". En condiciones normales, selecciona el siguiente token según esa distribución.

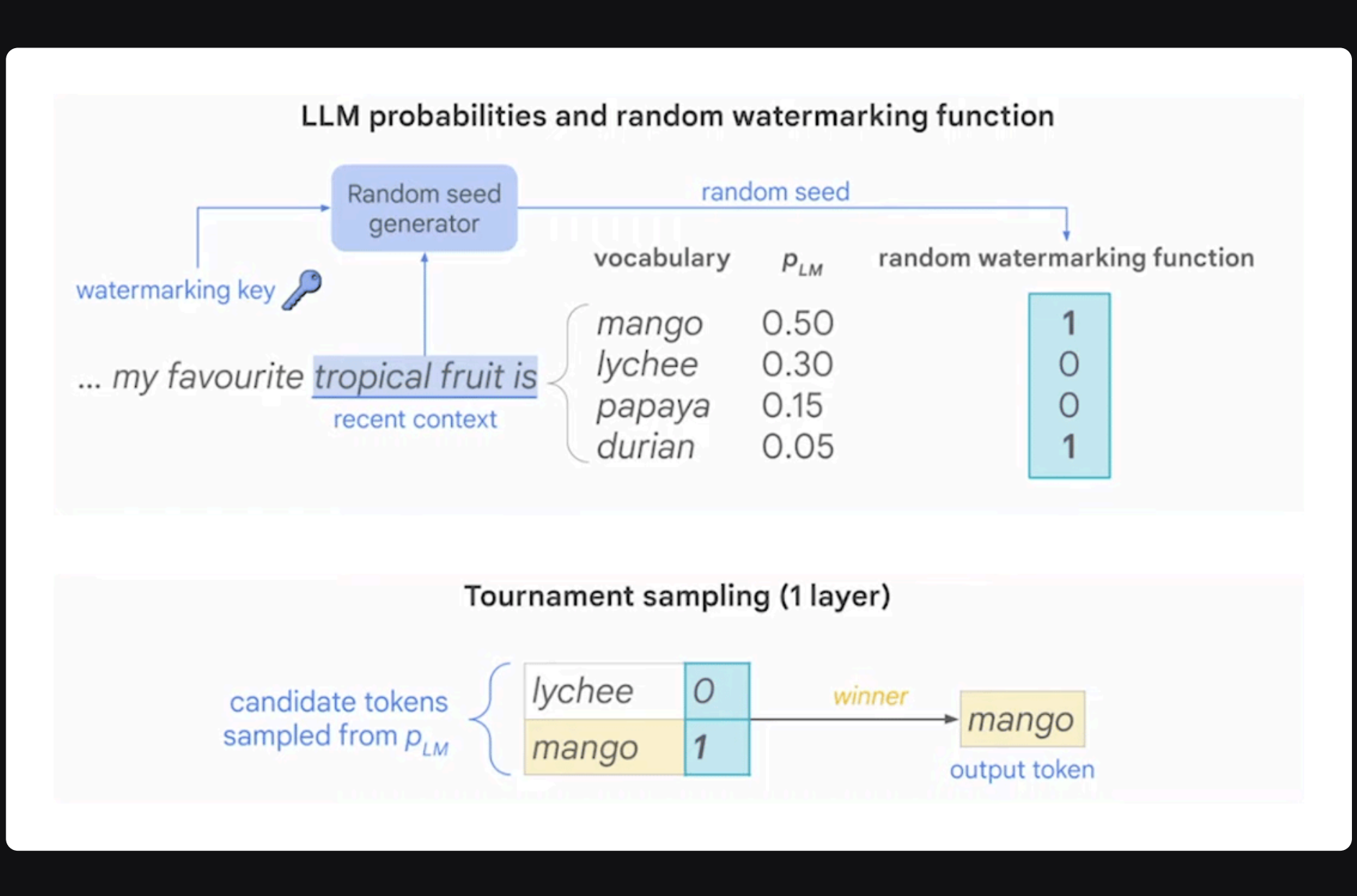


Ejemplo de asignación de probabilidades en un LLM

Con SynthID, Google modifica este proceso de forma casi imperceptible. ¿Cómo lo hace?

Primero, el modelo combina el **contexto más reciente** (por ejemplo, los últimos tres tokens: "tropical fruit is") con una clave secreta de watermarking (**watermarking key**).

Esa combinación genera una semilla aleatoria (**random seed**) única que se pasa a una **función de watermarking**. El resultado de esa función es un valor binario (1 o 0) asignado a cada token posible en el vocabulario, formando un patrón como 1-0-0-1.

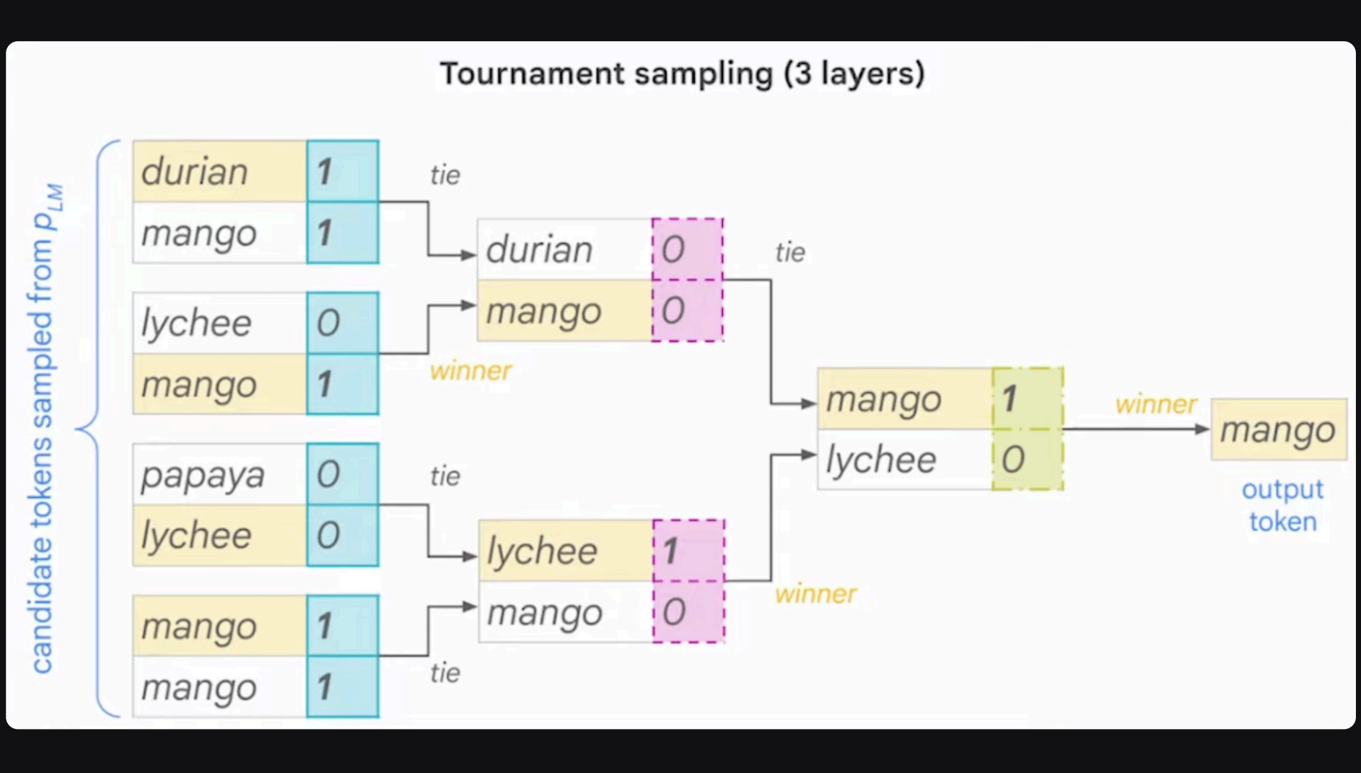


Función de watermarking en SynthID con Tournament Sampling

El modelo no elige ya un único token, sino que realiza un **tournament sampling**: en el que toma varios candidatos (normalmente más de dos, en tres capas sucesivas) y compara sus valores de watermark.

El token con valor de watermark = 1 tiene prioridad sobre los que tienen 0.

Si hay empate (1-1), se selecciona el token con mayor probabilidad en la distribución original. Ocurriría algo similar a lo que se ve en la imagen:



Tournament sampling en SynthID

Este pequeño ajuste **sesga ligeramente la distribución de tokens generados**, priorizando aquellos con watermark value igual a 1. El cambio es mínimo y no afecta ni al estilo ni al significado del texto, pero deja un patrón estadístico detectable.

¿Cómo se detecta texto generado por IA con Google SynthID?

Durante la fase de detección, se realiza el proceso inverso: se recorren los n-gramas del texto (grupos de tokens consecutivos) y, usando la misma clave secreta, se calculan los valores de watermark (1 o 0) esperados.

Luego se analiza la proporción de unos frente a ceros:

- Si el texto **no está marcado**, el número de 1s y 0s tiende a equilibrarse (~50/50).
- Si el texto **contiene un watermark**, habrá un exceso estadísticamente significativo de 1s.

Ese patrón basta para determinar, con alta fiabilidad, si un texto ha sido generado por IA o no.

Lo más interesante es que Google ha probado SynthID con y sin watermarking en **benchmarks** de rendimiento (fluidez, coherencia, calidad semántica) y los resultados son casi idénticos. Es decir, la **marca no altera la calidad del texto**, pero sí deja una firma invisible detectable.

Google tiene un notebook de ejemplo para que podamos probar el funcionamiento de SynthID en texto. Os recomiendo echar un vistazo también al repositorio de la implementación de SynthID basado en su paper.

Cómo funciona SynthID en imágenes: Cómo SynthID añade y detecta firmas en imágenes generadas por IA

En el caso de las imágenes, SynthID no trabaja a nivel de píxel, sino dentro del propio **espacio latente del modelo de difusión**, el lugar donde la red representa internamente los patrones visuales antes de decodificarlos en píxeles.

El espacio latente es la representación comprimida de una imagen dentro del modelo, una especie de "versión matemática" donde el modelo entiende estructuras, formas y texturas antes de renderizarlas.

Esto es clave porque **la marca de agua no se ve**, pero **queda incrustada en la estructura estadística de la imagen**. No altera colores (apenas), bordes ni detalles visibles, y sin embargo resiste a transformaciones como compresión, redimensionado o incluso filtros de color.

El proceso funciona así: durante la generación de la imagen, el modelo aplica una perturbación controlada sobre el vector latente, calculada a partir de una **clave secreta de watermarking**. Esa alteración mínima añade una firma digital única asociada al modelo, al usuario y a la sesión de generación a la imagen que ha creado el usuario.

Para la **detección**, una red neuronal complementaria (entrenada junto al generador) extrae esa firma del espacio latente inverso y determina si la imagen contiene marca o no.

Ejemplo de SynthID en imágenes: Aplicando una marca de agua

Veámoslo con un **ejemplo**. En mi caso, he creado dos notebooks disponibles en este gist que os comparto para que podáis probarlo:

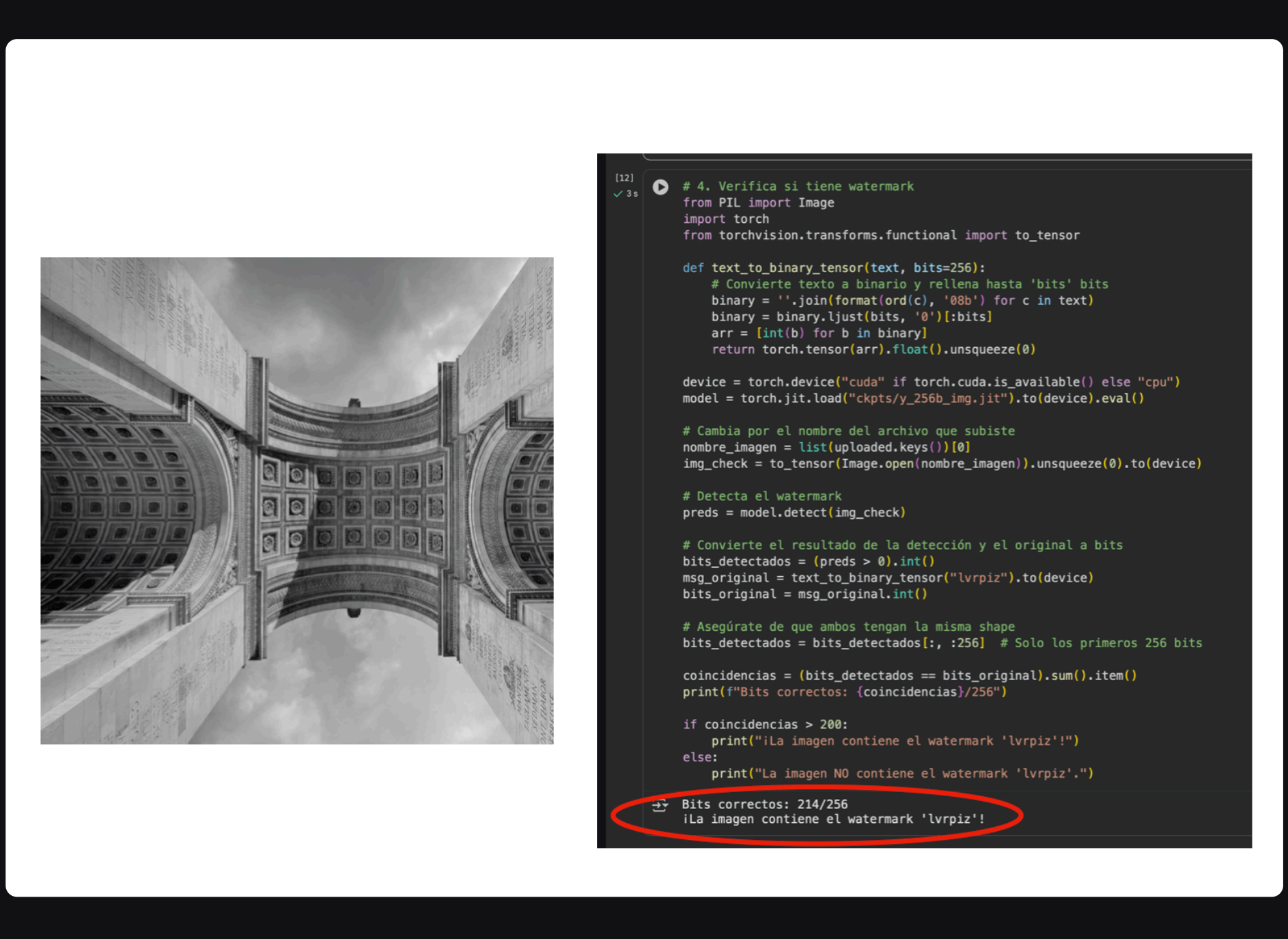
1. El primer notebook (watermark_embed.ipynb) **añade una marca de agua** a una imagen, simulando lo que haría Google al generar una imagen con modelos como Imagen 2 o Gemini Nano.
2. El segundo notebook (watermark_detect.ipynb): permite subir una imagen y **comprobar si contiene una marca de SynthID**.

Para que veáis cómo funciona, hice una prueba práctica en la que subí una fotografía del Arco del Triunfo y le añadí una marca de agua con el primer notebook. A simple vista, ambas imágenes son idénticas.

Comparativa de imágenes con watermark y sin watermark

Sin embargo, al pasarla por el segundo notebook, la detecta correctamente como watermarked.

Pero.. ¿qué pasa si la roto, la recorto y le aplico un filtro? El **resultado fue el mismo**: la firma siguió siendo detectada.



```
# 4. Verifica si tiene watermark
from PIL import Image
import torch
from torchvision.transforms.functional import to_tensor

def text_to_binary_tensor(text, bits=256):
    # Convierte texto a binario y rellena hasta 'bits' bits
    binary = ''.join(format(ord(c), '08b') for c in text)
    binary = binary.ljust(bits, '0')[:bits]
    arr = [int(b) for b in binary]
    return torch.tensor(arr).float().unsqueeze(0)

device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
model = torch.jit.load("ckpts/y_256b_img.jit").to(device).eval()

# Cambia por el nombre del archivo que subiste
nombre_imagen = list(uploaded.keys())[0]
img_check = to_tensor(Image.open(nombre_imagen)).unsqueeze(0).to(device)

# Detecta el watermark
preds = model.detect(img_check)

# Convierte el resultado de la detección y el original a bits
bits_detectados = (preds > 0).int()
msg_original = text_to_binary_tensor("lvrpiz").to(device)
bits_original = msg_original.int()

# Asegúrate de que ambos tengan la misma shape
bits_detectados = bits_detectados[:, :256]  # Solo los primeros 256 bits

coincidencias = (bits_detectados == bits_original).sum().item()
print(f"Bits correctos: {coincidencias}/256")

if coincidencias > 200:
    print("¡La imagen contiene el watermark 'lvrpiz'!")
else:
    print("La imagen NO contiene el watermark 'lvrpiz'.")
```

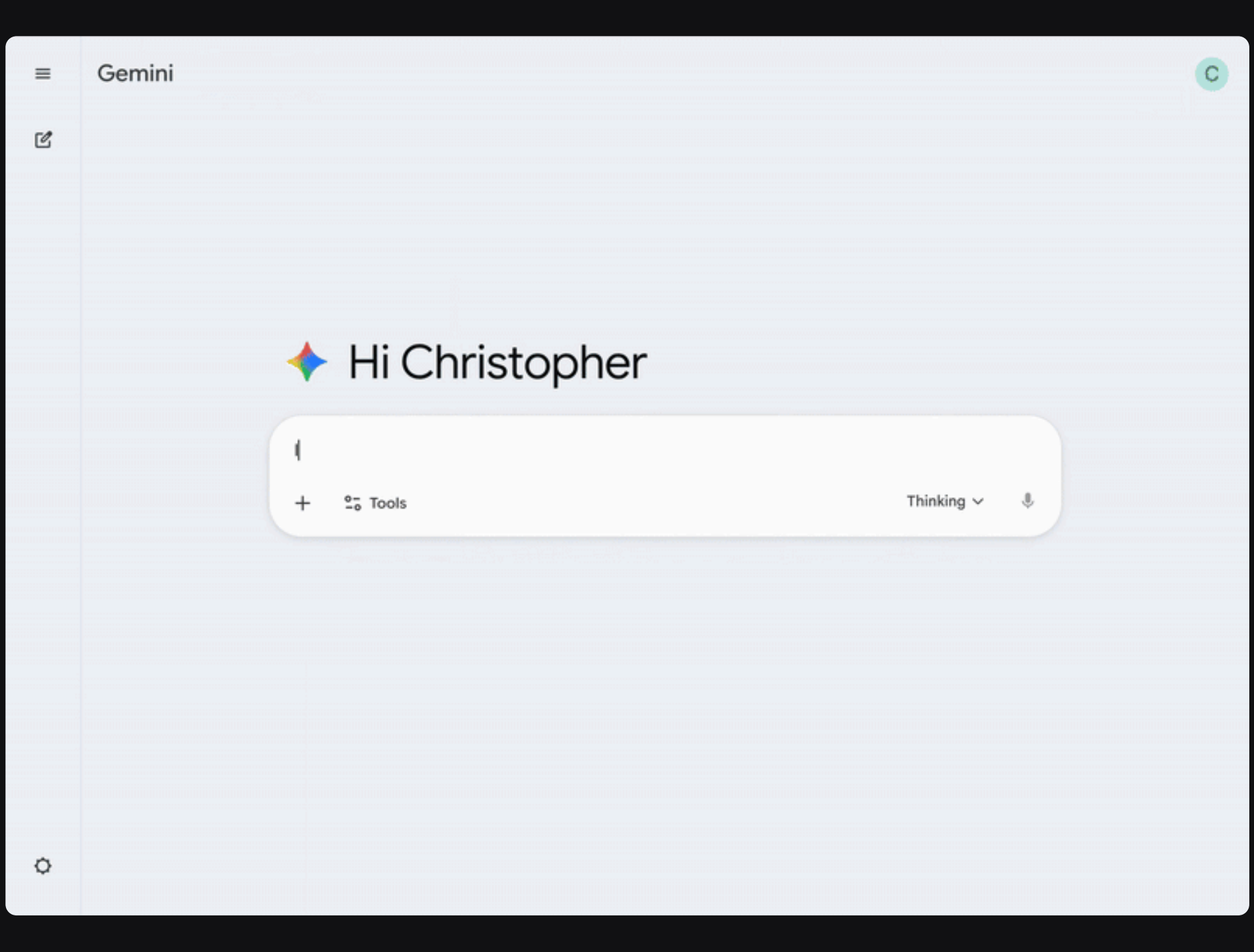
```
Bits correctos: 214/256
¡La imagen contiene el watermark 'lvrpiz'!
```

Detección del watermark en una imagen alterada

Esto demuestra lo más potente de SynthID: la marca no vive en los píxeles, sino en la geometría latente que da forma a la imagen. Aunque la modifiques, el patrón estadístico interno sigue ahí.

Te recomiendo probarlo con cualquier imagen: sube una, añádele el watermark y luego intenta eliminarlo aplicando transformaciones. Verás que, como ocurre con el sistema de Google, la detección sigue funcionando.

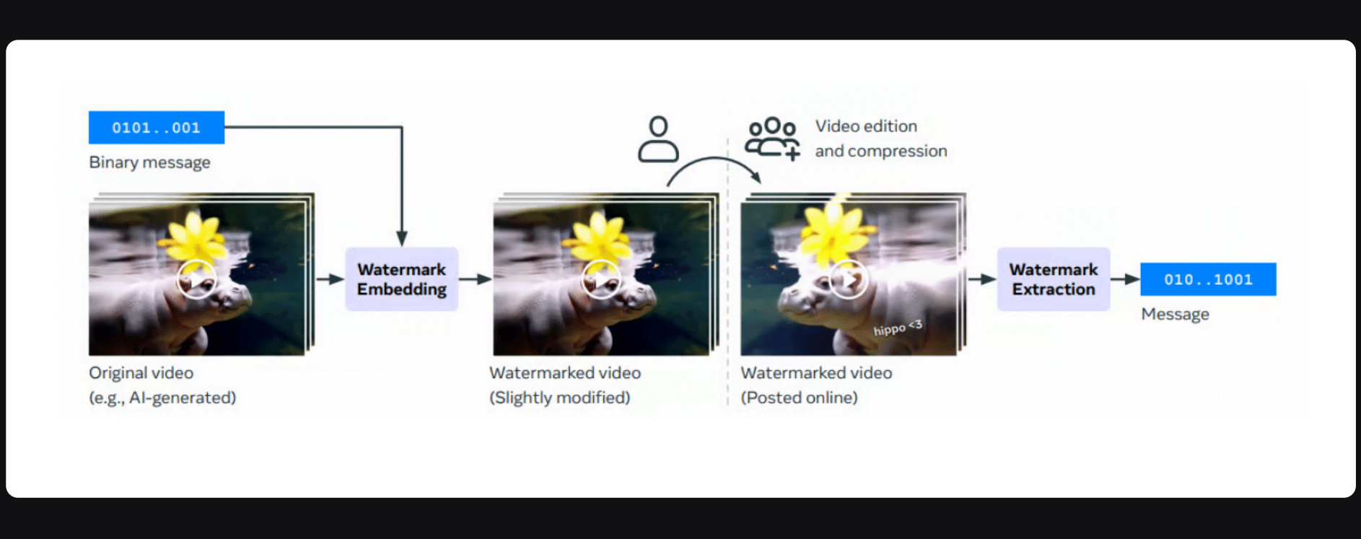
Con el lanzamiento de Gemini 3, Google ha integrado directamente en Gemini el detector de SynthID de tal manera que podemos subirle una imagen para detectar si ha sido o no generada por uno de sus modelos como en el siguiente vídeo:



Detección de imagen generada con IA en Gemini usando SynthID

Cómo funciona SynthID en vídeo: Watermarking fotograma a fotograma para contenido generado por IA

En vídeo, el sistema funciona exactamente igual que en imagen, pero aplicado fotograma a fotograma.

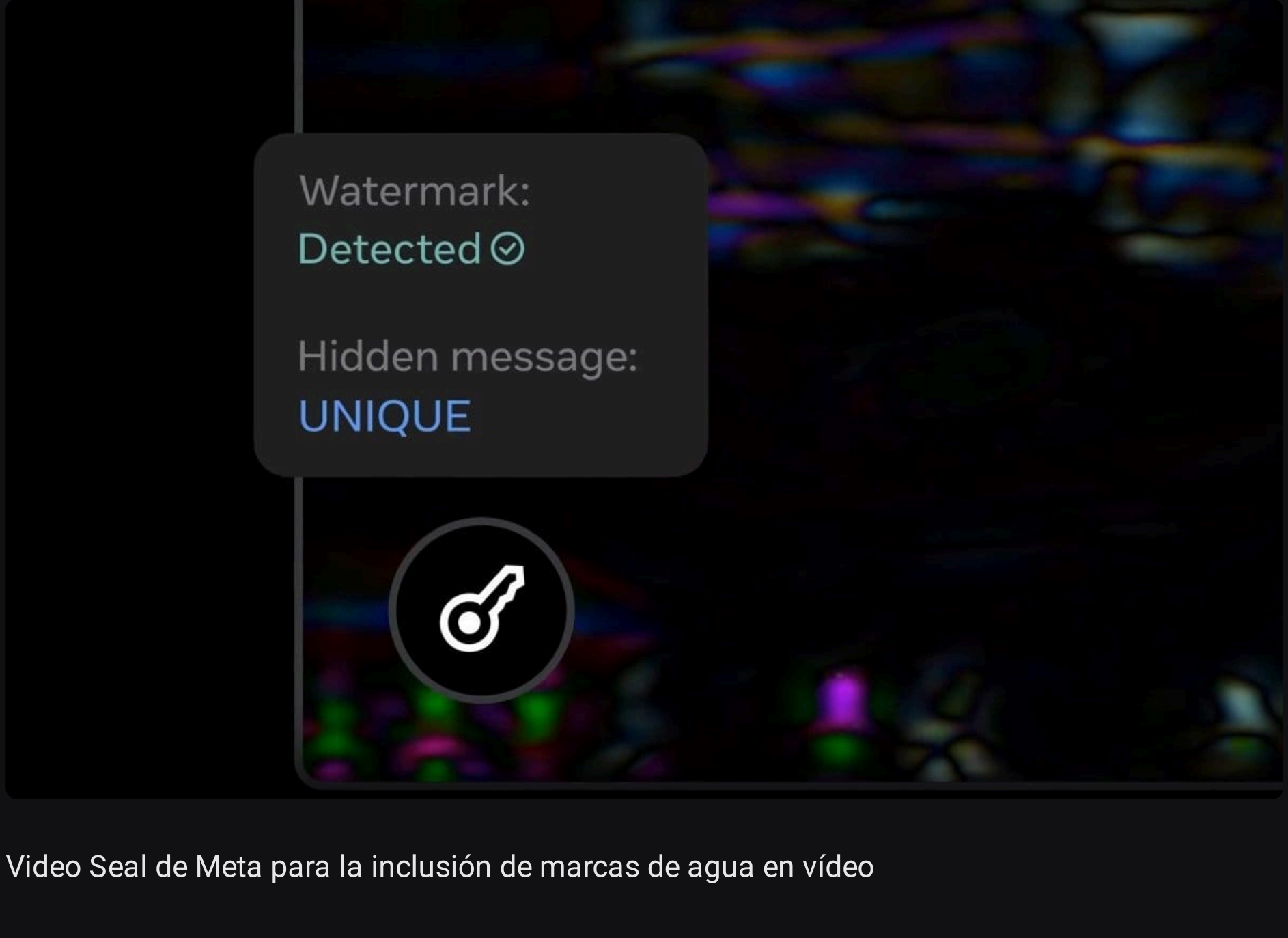


SynthID en vídeo fotograma a fotograma

Cada frame del vídeo se genera o procesa en su propio espacio latente, donde se inserta una pequeña perturbación controlada derivada de una **clave de watermarking**.

La clave del sistema está en cómo gestiona la coherencia entre frames. No basta con marcar cada fotograma por separado: si la marca variase demasiado, podría introducir inconsistencias detectables o perderse durante compresiones. SynthID soluciona esto **replicando el patrón estadístico con ligeras correlaciones entre fotogramas**, de forma que la firma se mantenga estable a lo largo del tiempo.

Durante la **detección**, el sistema analiza fragmentos del vídeo (por ejemplo, secuencias de 16 o 32 frames) y **reconstruye la huella latente combinando los valores de watermark detectados en cada uno**. El patrón resultante se compara con los valores esperados, obteniendo una probabilidad de marca presente en todo el clip.



Video Seal de Meta para la inclusión de marcas de agua en vídeo

Puedes probar a incluir marcas de agua en vídeo con la herramienta visual de Meta o directamente en un Notebook preparado basado en su paper.

SynthID es una de esas soluciones que parecen simples, pero que resuelven un problema enorme: distinguir entre lo que crea un humano y lo que crea una máquina.

Os recomiendo echar un vistazo a la presentación que compartí al principio del post que usé en la charla. Además de mostrar cómo funciona SynthID, explico varias formas de **modificar textos para que no sean detectados por los detectores de IA**, con ejemplos para que los pruebes.