

ViBe

Visual Background extractor

Antonio Cituk
Diego Durán
Héctor Pérez

ViBe briefly

ViBe es un algoritmo no paramétrico de sustracción de fondo. En lugar de usar modelos estadísticos, almacena muestras históricas por píxel y compara el valor actual con ellas.

Un píxel se clasifica como background si se parece a suficientes muestras previas, de lo contrario como foreground.

- Rápido
- Robusto al ruido
- Se adapta dinámicamente
- No necesita entrenamiento ni estimación de distribuciones

ViBe no estima el fondo con un único valor estadístico por píxel (media, mediana o una Gaussian mixture), sino con un conjunto de N muestras por píxel. Ese conjunto representa la variabilidad observada del píxel cuando estaba en el fondo. Es una aproximación no paramétrica y basada en muestras del proceso generador del fondo.

Inicialización del modelo:

- **Opción simple:** usar el *primer frame* y rellenar tomando valores del mismo píxel y de sus vecinos (muestreo espacial).
- **Opción más robusta:** usar varios frames iniciales y recoger muestras temporales y espaciales para cada píxel.

¿Cómo se calcula el modelo del fondo?

t: Tiempo en el que se toma la medición del píxel

x: Posición del píxel en la imagen (coordenadas)

p_t(x): Valor del píxel x en el tiempo t

n: Número de muestras almacenadas para cada píxel

R: Radio de la esfera en el espacio de color (RGB, HSV, etc.)

S_R(p_t(x)): Esfera de radio R centrada en p_t(x)

#min: Cantidad mínima de coincidencias necesarias para clasificar como fondo

—A—: Cardinalidad (número de elementos) del conjunto A

$$|S_R(p_t(x)) \cap \{p_1, p_2, \dots, p_n\}| \geq \# \text{ min} \Rightarrow \text{píxel es fondo (BG)}$$

Idea: Comparar el valor actual del píxel con las N muestras en S_R . Si se encuentra similitud con suficientes muestras es fondo, de lo contrario es objeto.

Regla de decisión (algoritmo)

1. Sea v_p el valor del píxel en el frame actual.
2. Contador `count = 0`.
3. Para cada muestra $s \in S_p$:
 - calcular distancia $d(v_p, s)$. Distancia típica:
 - escala de grises: $d = |v_p - s|$ (L1).
 - color: $d = \sqrt{(v_r - s_r)^2 + (v_g - s_g)^2 + (v_b - s_b)^2}$ (L2) o L1 en colores transformados.
 - si $d \leq R$ entonces `count += 1`.
 - si `count >= minMatches` → clasificar como **BG** (salida temprana).
4. Si al terminar `count < minMatches` → **FG**.

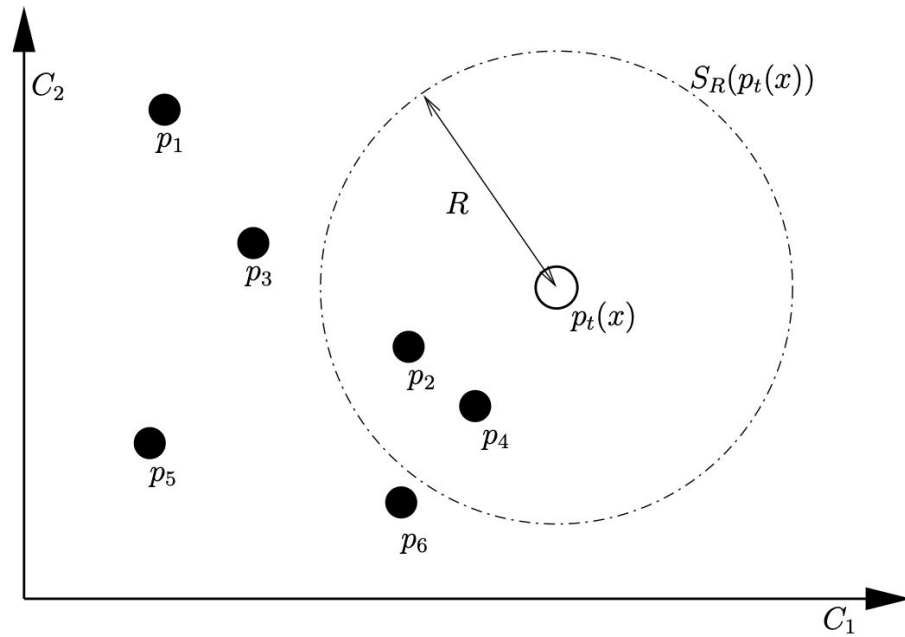
Parámetros claves:

- R (radio): controla tolerancia a variaciones. Valores típicos (grises) ~ 15–30; en color usar valores mayores o distancia euclidiana.
- #min: número mínimo de muestras que deben coincidir; típicamente 2 ó 3.
- Estrategia de *early exit*: se termina la búsqueda cuando se alcanzan #min, lo que reduce coste promedio.

Complejidad

- Peor caso por píxel: $O(N)$ comparaciones; promedio menor por early-exit.
- Total por frame: $O(W \cdot H \cdot \text{avg_cmp})$.

LA ESFERA S



Actualización del Modelo del Fondo

ViBe propone un enfoque más simple y efectivo:

reemplaza aleatoriamente una muestra cada vez que se actualiza el modelo.

- Esto genera un decaimiento exponencial en la probabilidad de que una muestra sobreviva en el tiempo.

Ventajas:

- Permite usar menos muestras sin perder capacidad adaptativa.
- El modelo se mantiene ágil y relevante.

$$P(t_0, t_1) = \left(\frac{n-1}{n} \right)^{t_1 - t_0} = e^{-\ln\left(\frac{n}{n-1}\right)(t_1 - t_0)}$$

Donde:

$P(t_0, t_1)$: probabilidad de que una muestra añadida en t_0 siga viva en t_1

n : número de muestras del modelo

Mecanismo principal

Cuando un píxel p es clasificado como **fondo**:

1. Con probabilidad $1/\phi$ (subsampling factor), elegir aleatoriamente un índice $i \in \{1..N\}$ y **reemplazar** $s_{p,i}$ por el valor actual v_p .
 - ϕ suele ser 16 (entonces probabilidad 1/16).
2. Con la misma probabilidad $1/\phi$, seleccionar un vecino aleatorio q (p. ej. en la vecindad 8-conectada) y actualizar uno de sus N samples reemplazándolo por v_p .

Actualizar sólo si está en fondo evita contaminar el fondo con objetos en movimiento.

La actualización **aleatoria** y la actualización a vecinos hacen que:

- Objetos que se detienen lentamente se integren al fondo (por difusión espacial).
- El modelo se adapta **gradualmente** (no hay reemplazo inmediato de todo el conjunto).

Consistencia Espacial

- ViBe también actualiza, al azar, un **vecino espacial** del píxel al que se le insertó una nueva muestra.
- Esto ayuda a mantener la **coherencia espacial** en:
 - pequeños movimientos de cámara
 - fondos que cambian lentamente (ej. hojas, sombras)

¿Por qué no arruina el modelo?

- Hay muchas muestras, así que el ruido ocasional no daña.
- Además, un valor incorrecto **no se propaga** si no coincide después.

Fortalezas y Debilidades

- **Movimiento de fondo**

ViBe maneja bien fondos que cambian lentamente (como ramas movidas por el viento), gracias a su **propagación espacial aleatoria** y actualización conservadora.

- **Ruido y compresión**

En pruebas con ruido artificial (PSNR hasta 30 dB), **ViBe mantiene buena precisión y recall**. EGMM se degrada más rápidamente ante el ruido.

- **Inicialización rápida**

Solo necesita **1 cuadro** para iniciar, usando vecinos espaciales para llenar el modelo. Desde el segundo frame, ya puede segmentar.

- **Bordes nítidos**

Los objetos del primer plano (FG) tienden a tener **bordes más definidos** que con EGMM.

- **Cambios de iluminación**

ViBe puede tolerar cambios suaves, pero **no tiene mecanismos específicos** para adaptarse a iluminación cambiante o abrupta.

- **Sombras**

No incluye técnicas para distinguir sombras del objeto. Puede confundirlas con parte del primer plano.

- **Determinismo**

Debido a su naturaleza aleatoria, **los resultados pueden variar** entre ejecuciones en la misma imagen.

Referencias

Barnich, O., & Van Droogenbroeck, M. (2009). *ViBe: A powerful random technique to estimate the background in video sequences*. In **Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)** (pp. 945–948). IEEE.
<https://doi.org/10.1109/ICASSP.2009.4959911>

Bouwman, T., Javed, S., Sultana, M., & Jung, S. K. (2014). *Background modeling methods in video analysis: A review and comparative evaluation*. **CVIU - Computer Vision and Image Understanding**, **122**, 1–29. <https://doi.org/10.1016/j.cviu.2013.12.005>