- Colisiones 2D
- Bloom

Voegeli Walter

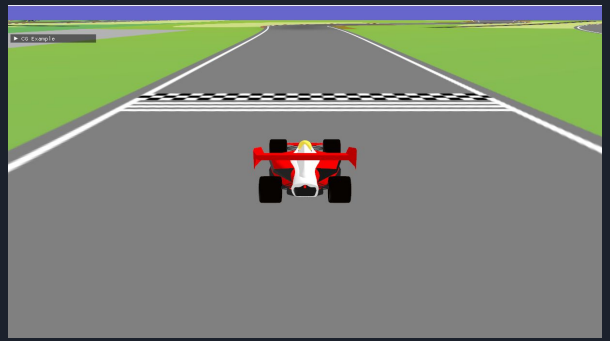
# INTRODUCCIÓN

- Se desarrolló un juego para poder aplicar los distintos métodos mencionados y luego poder hacer uso, modificar valores y poder observar los resultados.

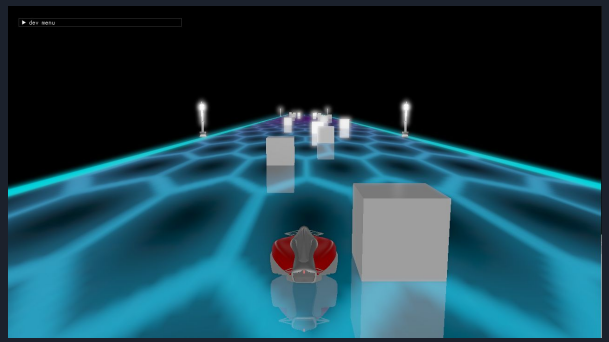
# INTRODUCCIÓN

- Se desarrolló un juego para poder aplicar los distintos métodos mencionados y luego poder hacer uso, modificar valores y poder observar los resultados.

- El juego es una modificación personal de un trabajo práctico a realizar en la clase de computación gráfica.



Trabajo original



Modificación

# ●Colisiones 2D

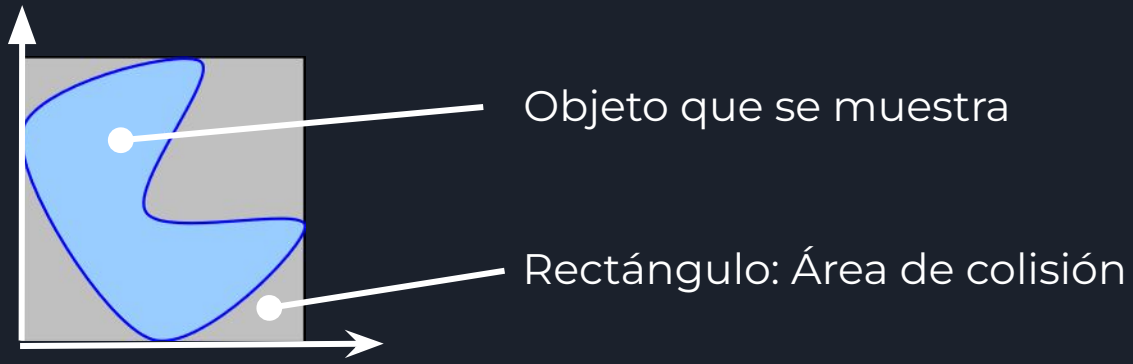
# Colisiones 2D

¿Cómo saber si un gameobject colisiona con otro en un ambiente de dos dimensiones?

# Colisiones 2D
# Algoritmo AABB

- AABB  es una simplificación de “Axis Aligned Bounding Box” (Caja delimitadora con ejes alineados)

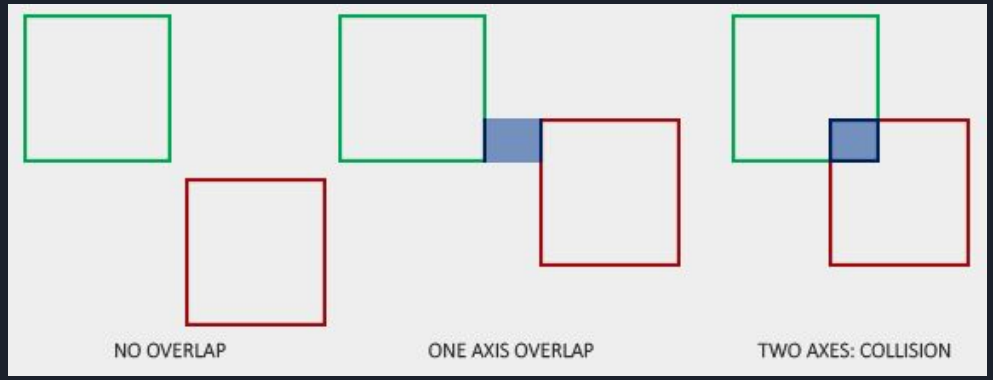
- El algoritmo establece un área de detección rectangular alineada con los ejes del objeto.



# Colisiones 2D
# Algoritmo AABB

- Luego verifica si la región delimitada del objeto se encuentra dentro de otra región de otro objeto.

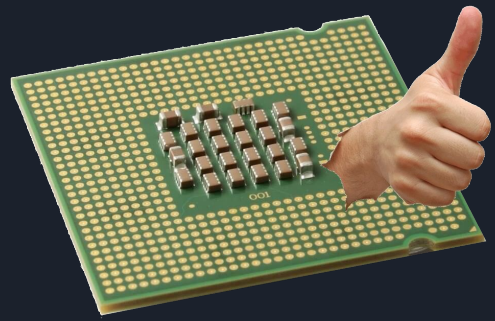
- Este procedimiento se logra comparando coincidencias en las posiciones x e y más el tamaño del AABB de cada objeto.



# Colisiones 2D
# Algoritmo AABB

VENTAJAS:

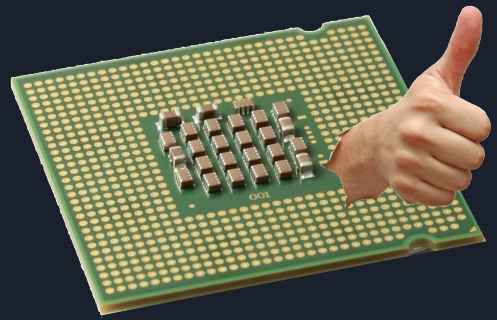
- Computacionalmente barato.
- Fácil implementación.



# Colisiones 2D
# Algoritmo AABB

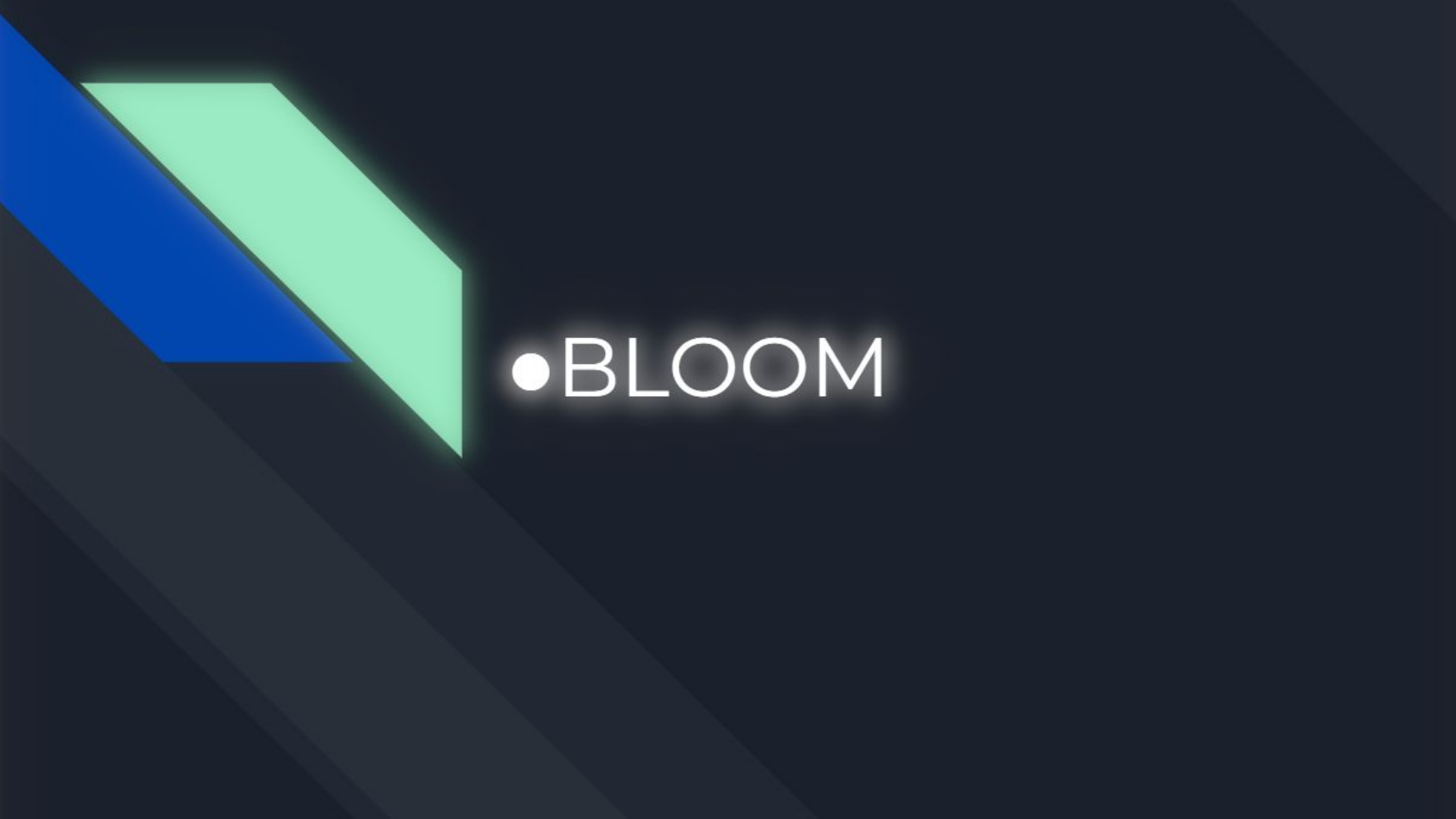
VENTAJAS:

- Computacionalmente barato.
- Fácil implementación.



DESVENTAJAS:

- Puede no representar correctamente el área de colisión del objeto.


BLOOM

# BLOOM *o resplandor.*

- Es un efecto gráfico que intenta simular como la luz en realidad tiende a invadir o algunos objetos a contraluz o continuos a un objeto emisivo muy brillante debido a las lentes de las cámaras.

*Escena de la película el señor de los anillos, donde se ve el brillo de la luz del sol invadir las montañas.*

# BLOOM En videojuegos.

- En los videojuegos, el bloom simula ese brillo adicional para lograr un efecto más “realista” ya que sería imposible lograrlo naturalmente debido al rango de brillo del monitor, además de dañar la vista.

- Si se excede una cantidad de bloom aplicado en la escena, puede perjudicar el realismo y la comodidad visual.

*Escena del juego the witcher 2*

# BLOOM En videojuegos

- Gracias a este efecto, se puede percibir con mayor claridad la diferencia de intensidad de iluminación y brillo de distintos objetos en videojuegos.

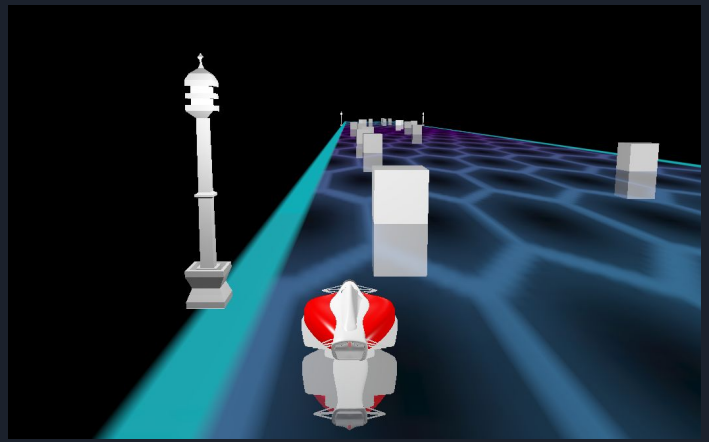
BLOOM ACTIVADO

BLOOM DESACTIVADO

# BLOOM Pasos del algoritmo

- Para implementar el efecto Bloom, se renderiza la escena normalmente y se extrae en una textura aparte las escenas con mayor brillo.

PASO 1

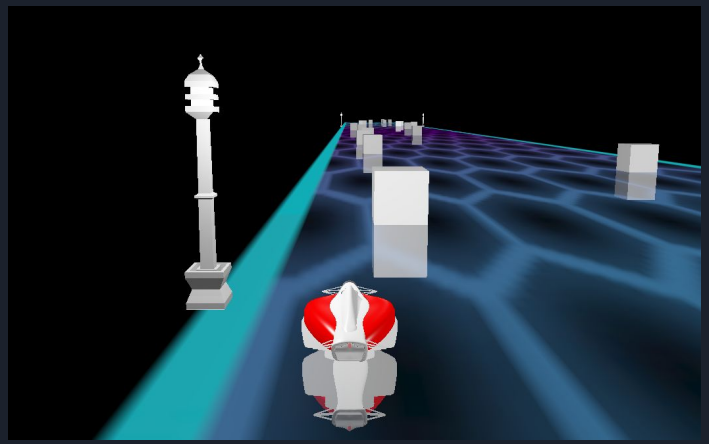


Render normal con iluminación phong.

# BLOOM Pasos del algoritmo

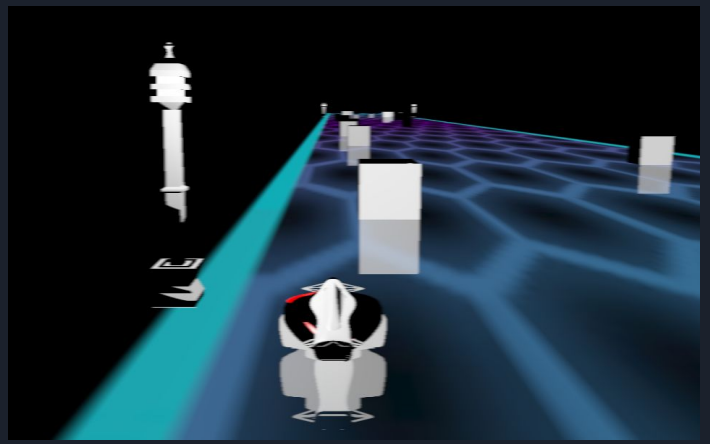
- Para implementar el efecto Bloom, se renderiza la escena normalmente y se extrae en una textura aparte las escenas con mayor brillo.

PASO 1



Render normal con iluminación phong.

PASO 2



Extracción de los fragmentos más brillantes.

# BLOOM

Pasos del algoritmo - Extracción del brillo.

- Para poder obtener mejores resultados en la extracción del brillo, se renderiza a una profundidad de colores más alta para obtener un rango mucho mayor de brillo del que podamos acotar.

- Opengl por default establece un rango de 0 a 1 de valores de brillo, esto se conoce como LDR (low dynamic range)

Izquierda: LDR

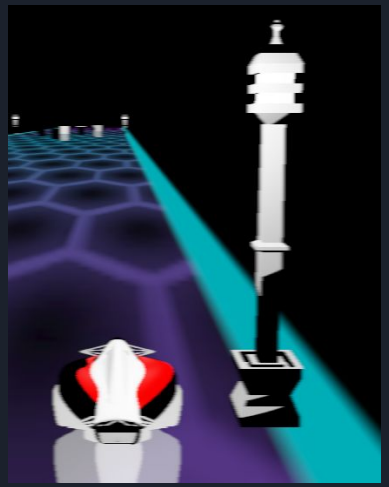
Derecha:
High dynamic Range

# BLOOM

Pasos del algoritmo - Extracción del brillo.

- Utilizando HDR podemos aumentar los detalles y el rango de brillo más allá de 1 y, de este modo, establecer mejor las regiones que se aplica bloom.

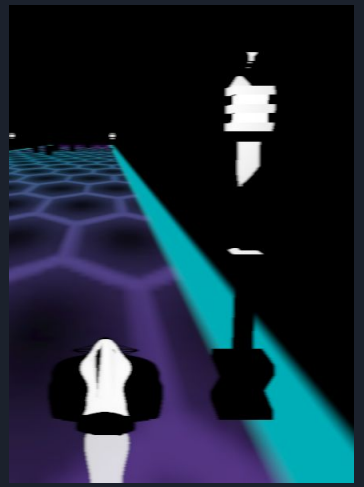
- Debido a que los monitores muestran brillo de 0 a 1, hay que regresar HDR a LDR, esta transformación se llama tone mapping.



Extracción de brillo

**Izq:** LDR, menos control.

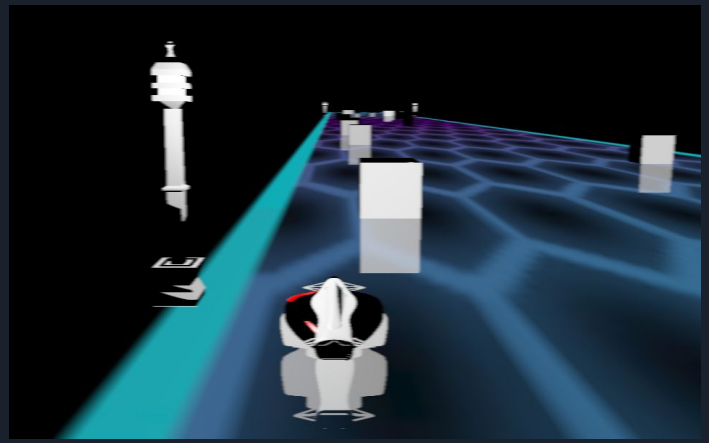
**Der:** HDR, mayor control.



# BLOOM Pasos del algoritmo

- Una vez almacenada la textura obtenida en el paso 2, se le aplica desenfoque o difuminación también conocida como gaussian blur.

PASO 2



Textura con fragmentos brillantes.

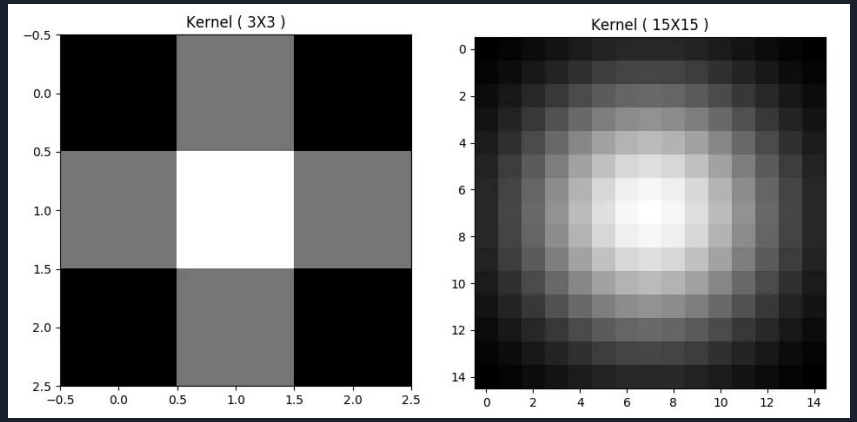
PASO 3

Blur aplicado a la textura

# BLOOM Pasos del algoritmo - Difuminado

- El método más conocido para realizar blur es gaussian blur por su estilo natural de difuminado, es un algoritmo fácil de implementar pero puede llegar a ser muy costoso computacionalmente.

- Consta de definir y utilizar una matriz cuadrada (kernel) con diferentes pesos (valores), conteniendo el mayor en su centro.

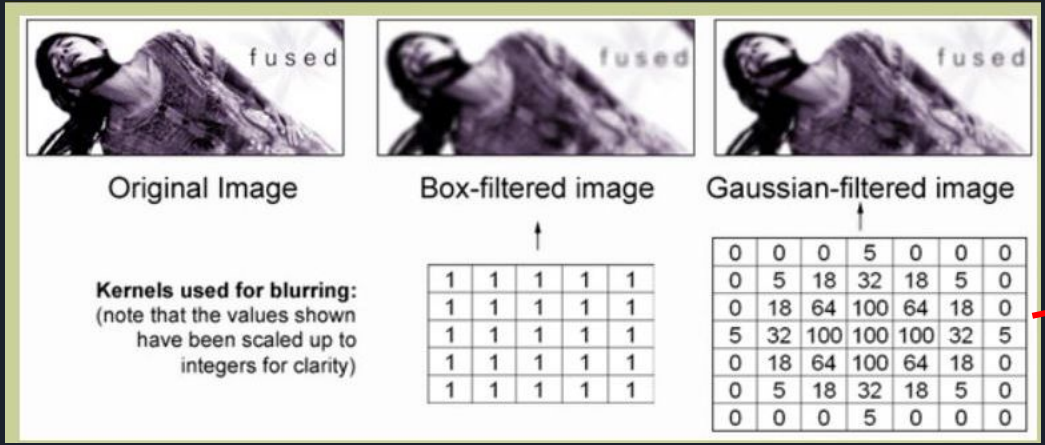


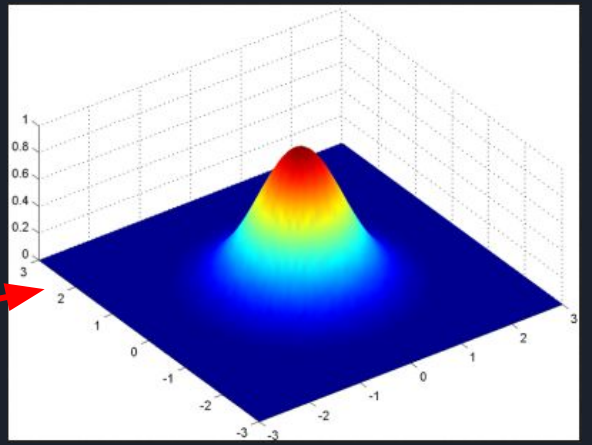

Diferentes tamaños de kernel.

# BLOOM Pasos del algoritmo - Difuminado

- Gaussian blur multiplica dicho kernel por cada texel, centrando el mayor peso (el centro de la matriz) en el texel que se difumina hacia los vecinos.

- Gracias a la distribución gaussiana de pesos, a medida que se alejan los texeles vecinos del centro, menos influencia de su color tendrá.

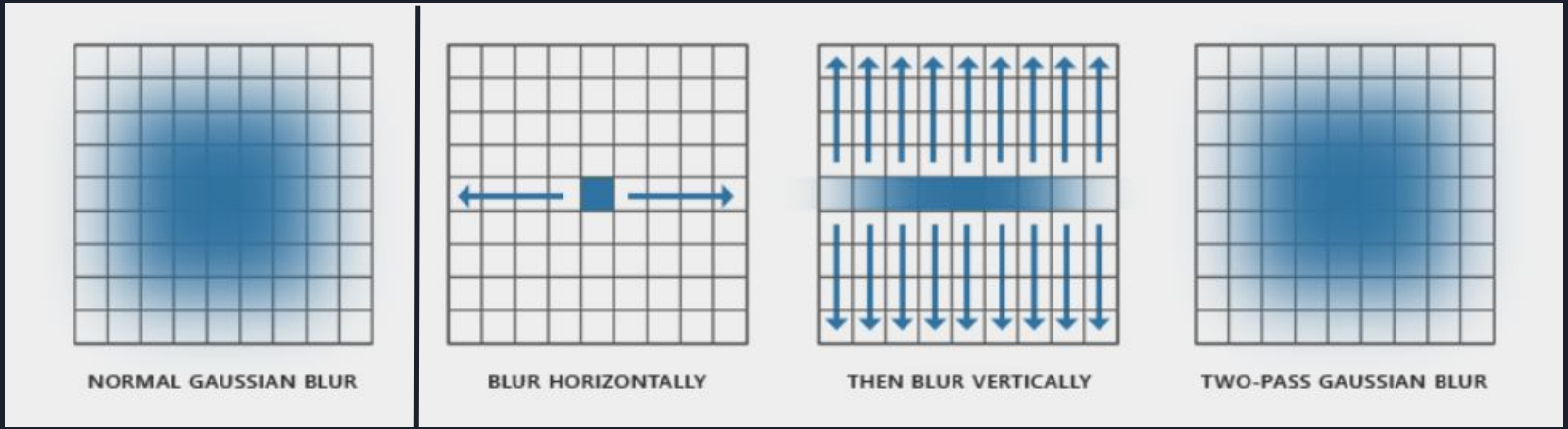


# BLOOM Pasos del algoritmo - Difuminado

- Afortunadamente se puede reducir el coste computacional separando la ecuación de dos dimensiones en dos ecuaciones de una dimensión.

- Primero se realiza un blur horizontal y luego a dicho resultado se le aplica un blur vertical, este método se conoce como **two-step gaussian blur.**



# BLOOM

## Pasos del algoritmo - Difuminado

Siendo una textura de 480.000 tx (800x600)

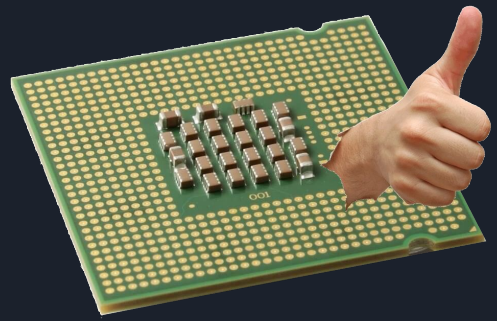
- Con gaussian blur y un kernel de dim 15 x 15 se requieren 108.000.000 cálculos .
- Con este método de gaussian blur y dos kernels de dim 15 x 1 se requieren ¡14.400.000 cálculos!

# BLOOM Pasos del algoritmo - Difuminado

Siendo una textura de 480.000 tx (800x600)

- Con gaussian blur y un kernel de dim 15 x 15 se requieren 108.000.000 cálculos .

- Con este método de gaussian blur y dos kernels de dim 15 x 1 se requieren ¡14.400.000 cálculos!

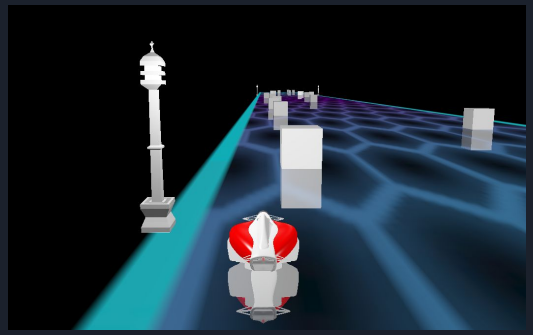


Usá **two-step gaussian blur** :')

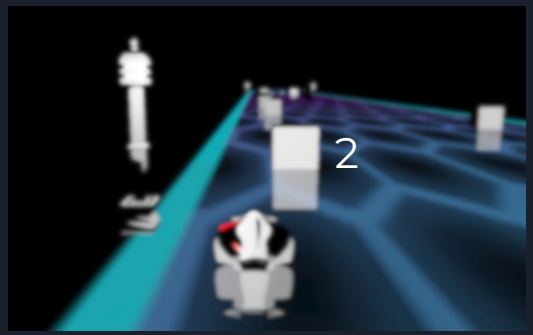
# BLOOM Pasos del algoritmo

- Finalmente se combinan las texturas, tanto la normal como la textura de los fragmentos brillosos difuminada.

Textura normal.



Textura blureada.



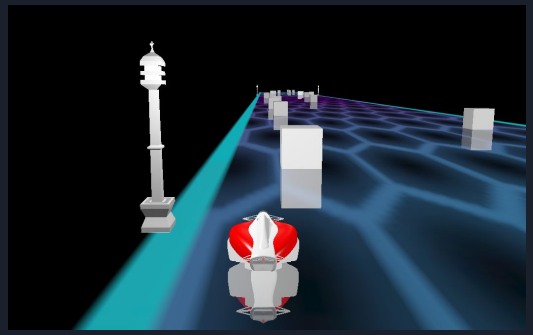
PASO 4

Combinación

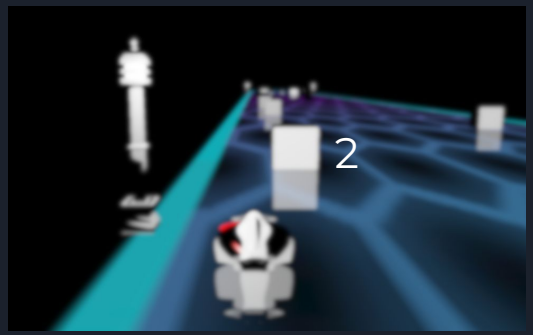
# BLOOM Pasos del algoritmo

- Finalmente se combinan las texturas, tanto la normal como la textura de los fragmentos brillosos difuminada.

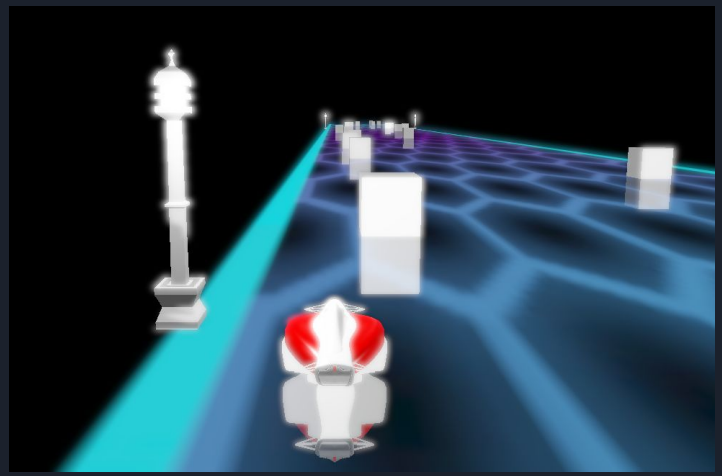
Textura normal.



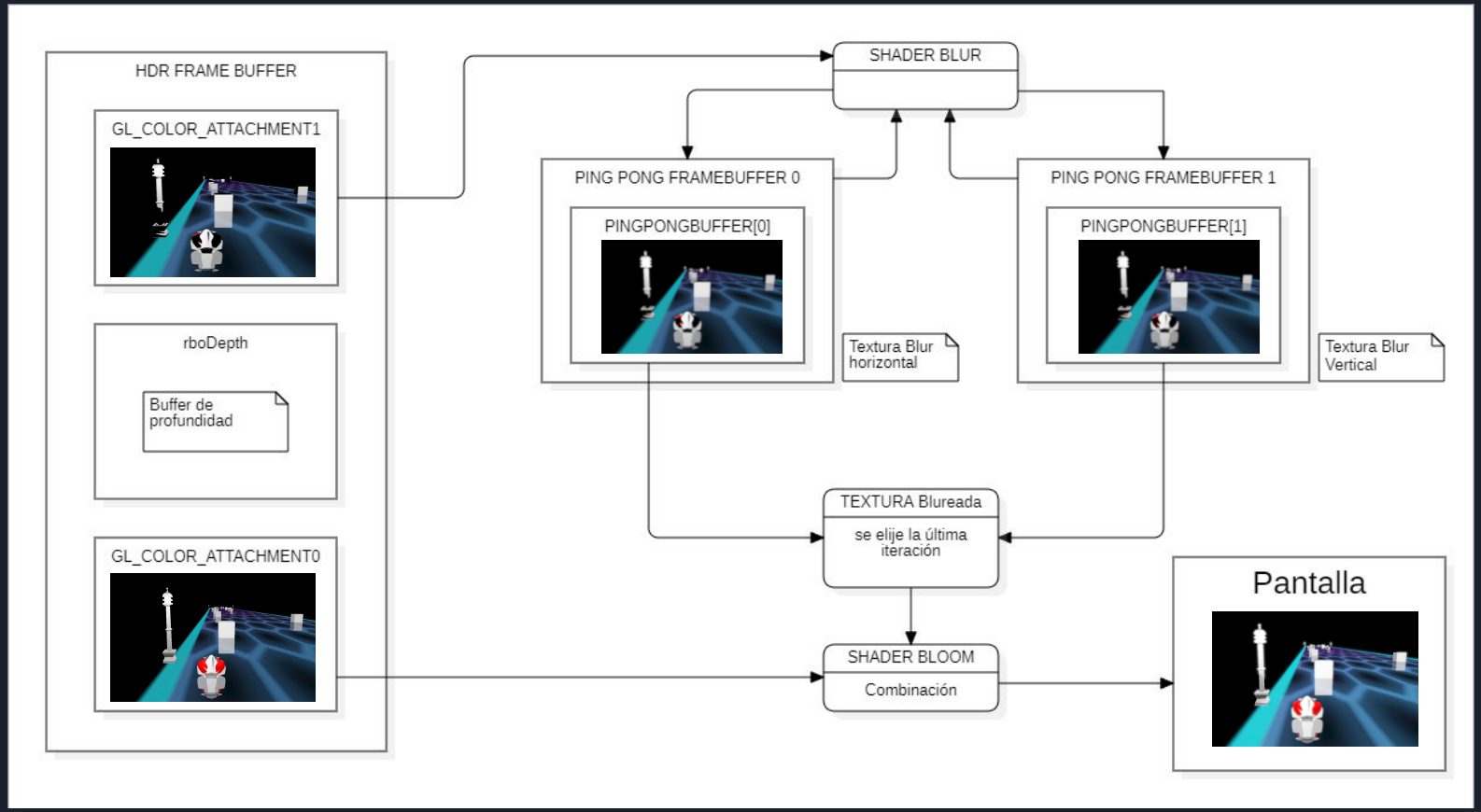
Textura blureada.



Resultado



# BLOOM Opengl, flujo de datos.



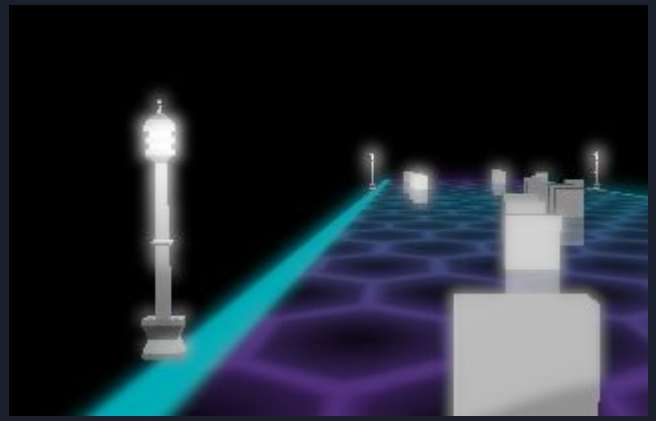
# BLOOM Resultado

## SIN BLOOM

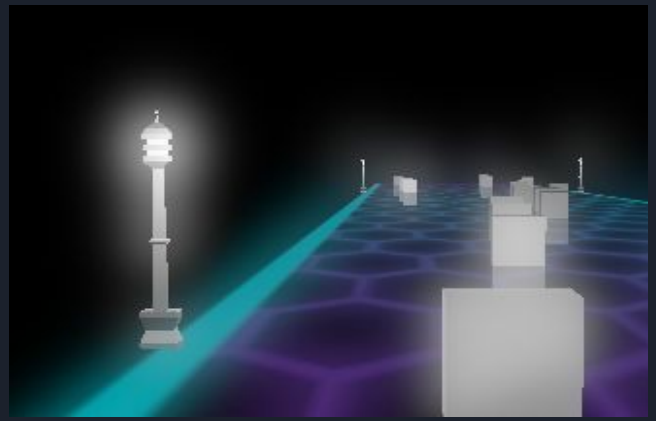

## CON BLOOM

# BLOOM

- El bloom en teoría no es una técnica complicada, lo más complicado es encontrar los parámetros ideales para no “quemar” la imágen y que quede visualmente atractivo.
- Por lo general la calidad visual del bloom depende de la cantidad de blur y el rango del brillo a filtrar (mejora con HDR).
- Debido a que el blur se ejecuta en una textura, el costo depende de la resolución.



6 Iteraciones de blur



100 iteraciones de blur

# GRACIAS POR SU ATENCIÓN