

Esquemas para la administración de múltiples cámaras en aplicaciones de Realidad Aumentada

**Mauro Agustín
Esposito**



Facultad de Informática
Universidad Nacional de La Plata

Contenidos

- Introducción
- Background
 - Conceptos básicos de Realidad Aumentada.
 - Tecnologías aplicadas.
- Librería desarrollada
 - Limitaciones de la Realidad Aumentada.
 - Estructura de la librería.
- Caso de estudio
 - Incorporación de la librería a una aplicación convencional de R. A.
- Conclusiones y trabajo a futuro

Introducción

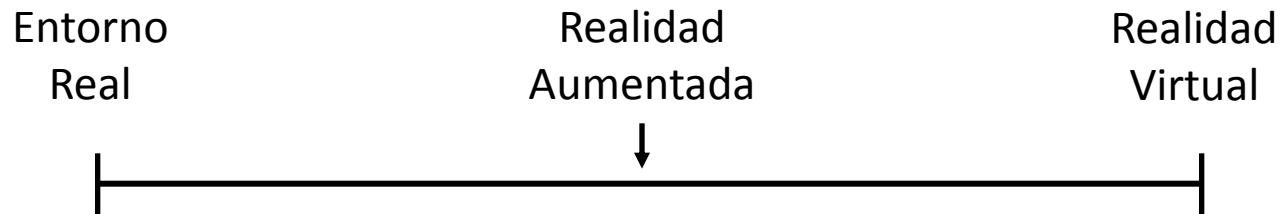
La Realidad Aumentada como una nueva forma de comunicar.

Diversas aplicaciones, orientadas al entretenimiento, marketing, presentaciones corporativas, entre otros.

Distintas maneras de interactuar y de aprovechar sus contenidos.

Realidad Aumentada

Concepto de Realidad Aumentada



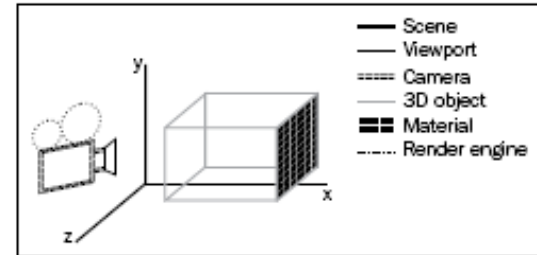
Plataformas de la Realidad Aumentada. Elementos.

Ejemplos:

- Computadoras de escritorio y notebooks
- Palms y smartphones
- Head Mounted Displays

Elementos de la Realidad Aumentada

➔ Escena y cámara 3D



➔ Patrón a reconocer. Proceso de detección



➔ Modelo



Tecnologías

FLARToolkit

- Codificado en ActionScript (Flash).
- Requiere una librería para poder renderizar gráficos 3D.

Papervision3D

- Motor de renderizado dinámico sobre Flash. Basado en líneas y puntos
- Primitivas. Soporte de modelos externos. Manejo de cámara, escena y proyección.
- **Maneja el evento de entrada a un nuevo frame → Clase AbstractView**

Adobe AIR

- Crear ejecutables a partir de código de lenguajes para RIA (Javascript, ActionScript, HTML)
- Acceso seguro a los recursos del usuario

Problema e hipótesis

Limitaciones de la Realidad Aumentada en plataforma *desktop*

Otorga una perspectiva estática e invariable del patrón.

No permite tener una apreciación completa del modelo, desaprovechando funcionalidad

Se encuentra en desventaja frente a otras plataformas.

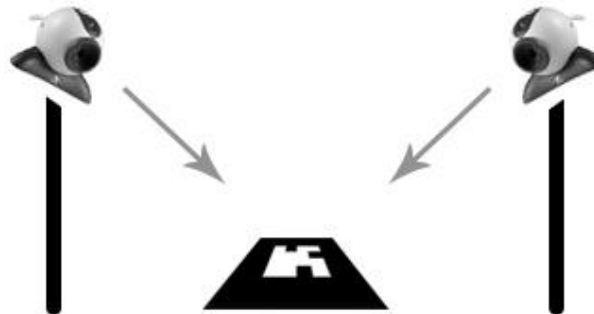
Ejemplo: Museo de Ciencias Naturales Vs. Aplicación online de una aerolínea.



Problema e hipótesis

Hipótesis de trabajo

Colocar más de una cámara web, en distintos puntos del espacio, apuntando al marker simultáneamente.



Interrogantes

- ¿Es posible escalar una aplicación de RA a más de una cámara?
- ¿Hay una manera de administrar todos los videos en tiempo real?
- ¿Se verá afectada la performance de la aplicación?

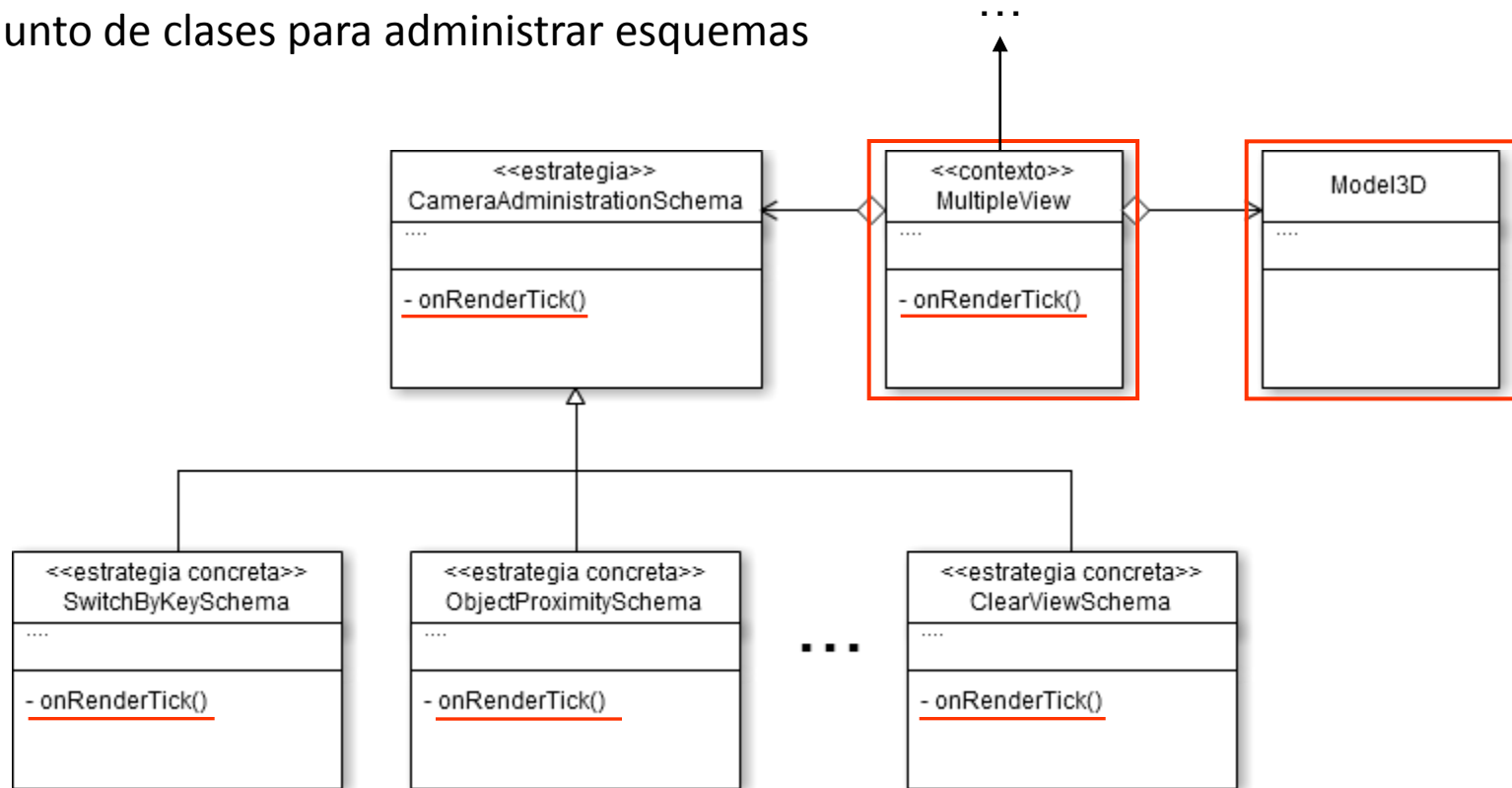


Librería desarrollada

- ➔ Desarrollo de una librería que permita conectar más de una cámara y los flujos de video de cada una de ellas.
- ➔ Diseño de diversos esquemas para administrar esos flujos.
- ➔ Posibilidad de instanciar un esquema en particular, según sea mas apropiado para el comportamiento de la aplicación.

Librería desarrollada

Conjunto de clases para administrar esquemas



Patrón **Strategy** para el manejo de los esquemas.

Librería desarrollada

onRenderTick():

Es aquí donde se concentra la lógica del esquema. Se decide que cámara va a estar activa durante el próximo frame.

Es delegado al esquema concreto.

Si se quisiera agregar un nuevo esquema, bastaría con definir una clase que extienda a *CameraAdministrationSchema* e implemente dicho método.

Esquemas de coordinación

Esquema manual

Las cámaras se intercambian manualmente.
No es influido por ninguna variable externa.

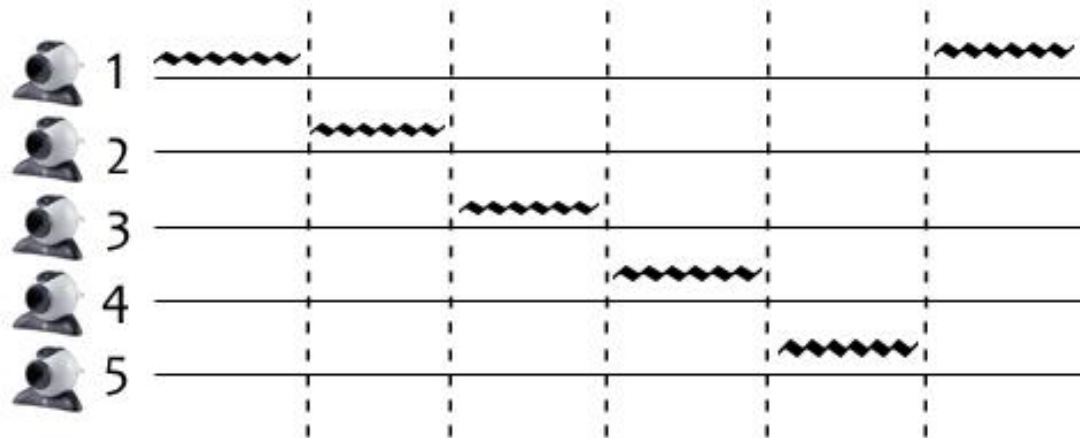


Esquemas de coordinación

Esquema de quantum o slice de tiempo

Cada cámara estará activa mediante un cierto período de tiempo.

Se chequea un *timer* y cuando este alcanza el valor configurado, se cede el control a la siguiente cámara y se vuelve a cero.

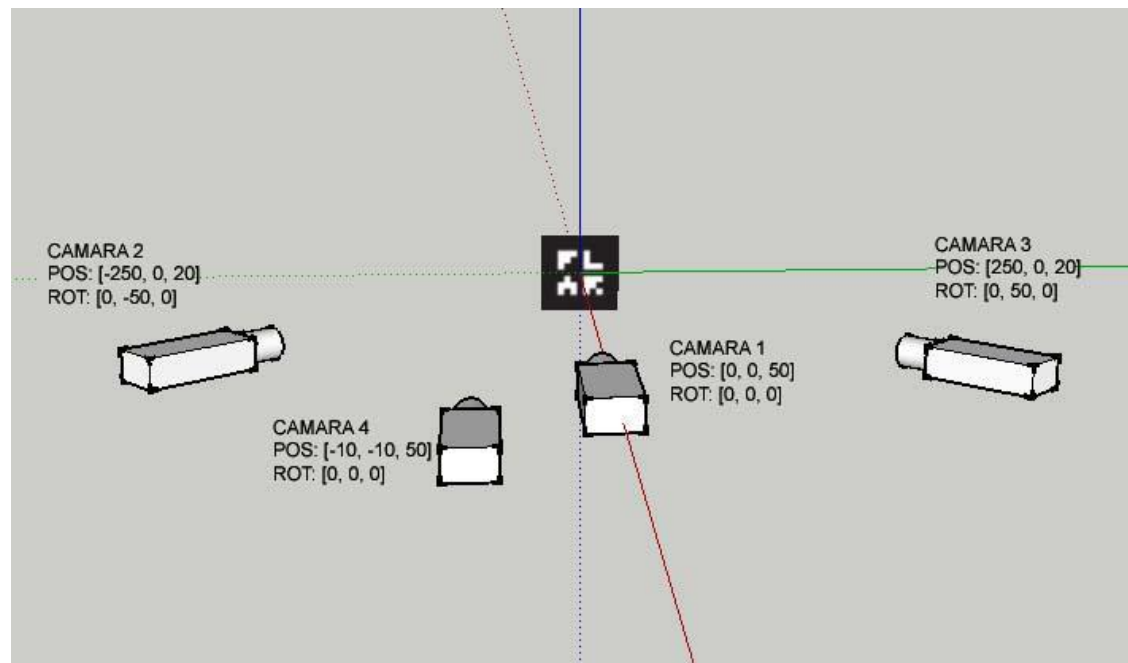


Esquemas de coordinación

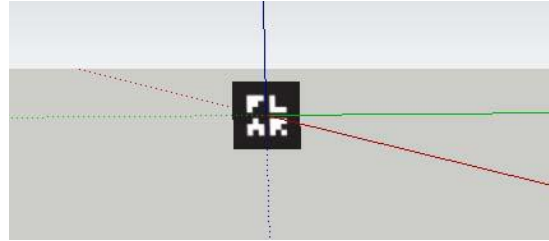
Esquema de proximidad al modelo

Permite individualizar un objeto dentro del modelo 3D para simular que esta siendo “seguido” por las cámaras.

Es necesario conocer la distancia entre cada cámara y el objeto en cuestión.

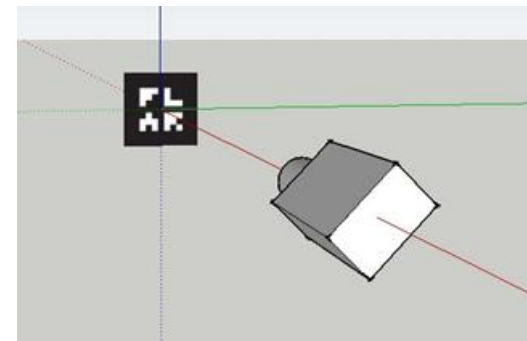
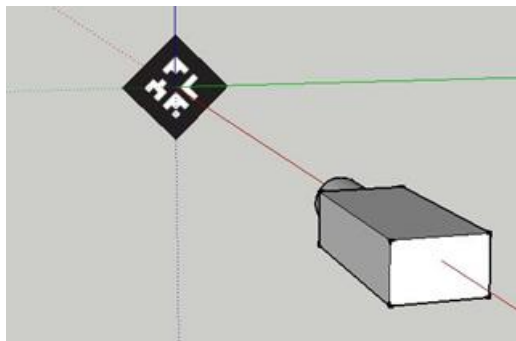


Esquemas de coordinación



Matriz de transformación

Almacena las dimensiones, posición y orientación del patrón.

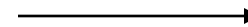


Esquemas de coordinación

Matriz de transformación

Interesa conocer la distancia real entre la cámara y el patrón.

Es invertida para obtener la posición y rotación de la cámara con respecto al patrón.



Posición = [$x = m14$, $y = m24$, $z = m34$]

Rotación = *matrixToEuler(matriz)*

Distancia entre la cámara y el marker:

$$d_{AB} = \sqrt{(x_B - x_A)^2 + (y_B - y_A)^2 + (z_B - z_A)^2}$$

$$\begin{vmatrix} m11 & m12 & m13 & m14 \\ m21 & m22 & m23 & m24 \\ m31 & m32 & m33 & m34 \\ m41 & m42 & m43 & m44 \end{vmatrix}$$

Esquemas de coordinación

En cada frame se calcula la posición de cada cámara y se escoge la menor

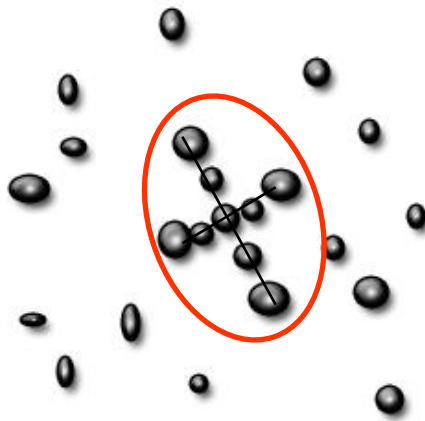
50	82.3	82.3	65.7
0	1	2	3

Esquemas de coordinación

Esquema de proximidad a un evento

Escoge la cámara que esta más próxima a un evento en particular.
Hace uso de los mismos cálculos del esquema anterior.

Se debe definir un evento de interés, por ejemplo, 10 partículas se encuentran muy cerca y componen un nuevo objeto.



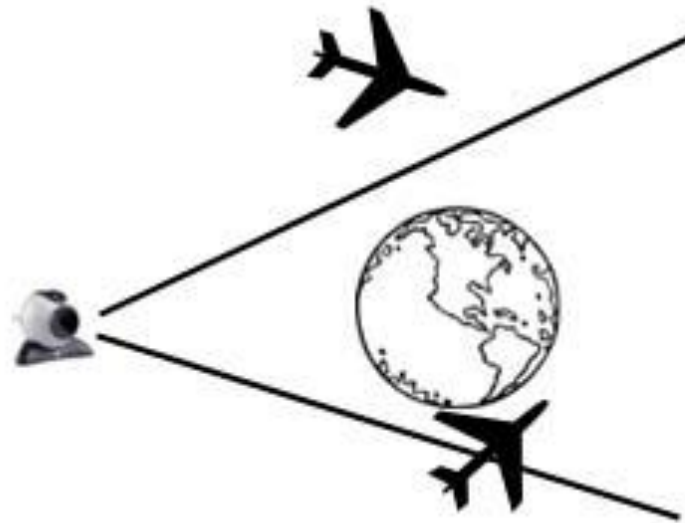
Esquemas de coordinación

Esquema de mejor perspectiva o clearview

Selecciona la cámara que tenga una vista completa del modelo.

Si existe más de una candidata, escoge una al azar.

Objetos dentro y fuera del ángulo de vista de la cámara → *Culling*



Elementos y escenario de trabajo

Adaptación de una aplicación convencional.

Marker cuadrado de 8cm. de lado.



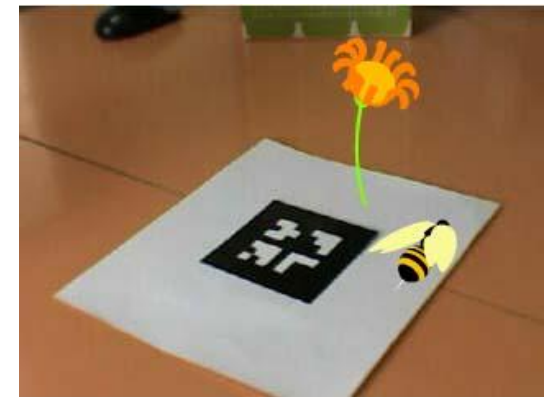
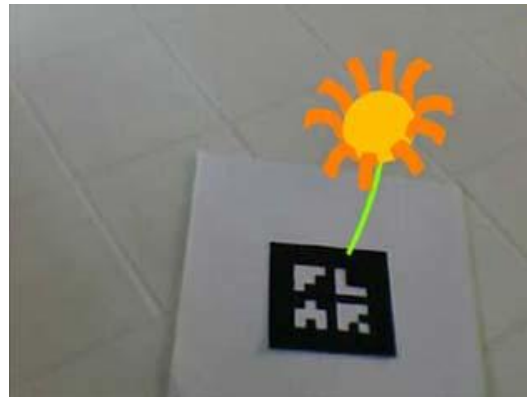
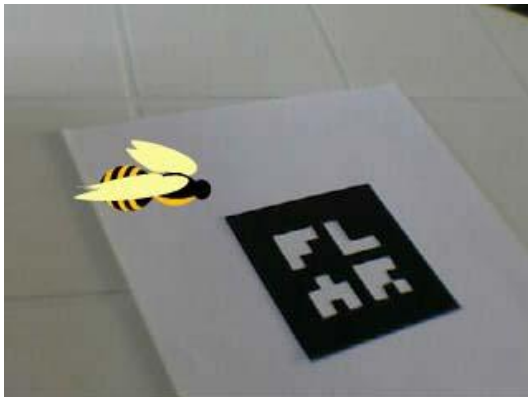
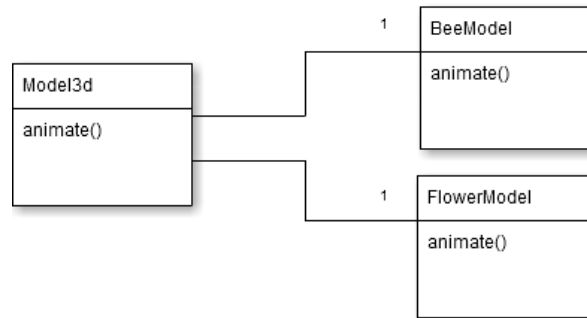
4 cámaras:

- Genius I-LOOK300
- Logitech QuickCam Express 100
- Logitech QuickCam Express 150
- Creative Laptop Integrated Webcam



Composición del modelo

Objeto compuesto por dos clases que importan modelos 3D.



Evaluación de los esquemas

Esquema manual

Tecla C => cambio secuencial. Posibles flujos inadecuados

Números 1-4 => cambio directo

El cambio es inmediato.

Evaluación de los esquemas

Esquema de quantum o slice de tiempo

Se debe especificar el valor del slice antes de la ejecución.

El usuario no interviene una vez que la aplicación comienza su ejecución.

El cambio es inmediato.

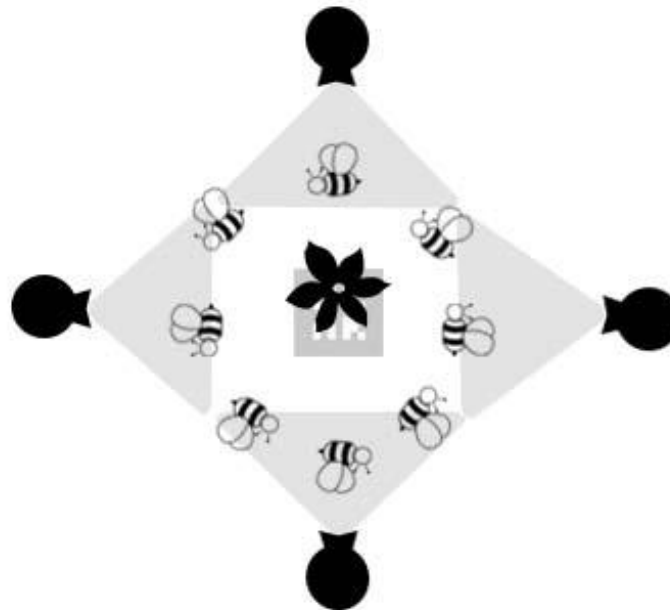
Evaluación de los esquemas

Esquema de proximidad al modelo

Cámaras dispuestas rodeando al marker

La cámara activa es aquella que se encuentre más próxima a la abeja.

El cambio se demora debido a los cálculos que se requieren.



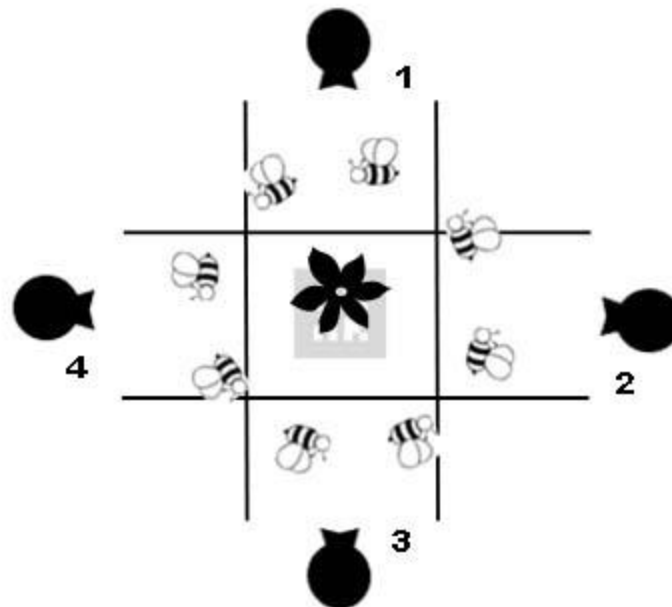
Evaluación de los esquemas

Esquema de mejor perspectiva (clearview)

Cámaras dispuestas rodeando al marker

En cada instante hay dos potenciales cámaras activas.

El cambio se demora debido a los cálculos que se requieren.



Resultados obtenidos

- Aplicaciones de RA en desktop con más de una cámara.
- Esquemas implementados para su manejo y otros propuestos.
- Librería para adaptar aplicaciones convencionales a n cámaras funcionando simultáneamente.
- Posición de la cámara dentro del espacio a partir de los datos obtenidos del patrón detectado.

Aspectos a tener en cuenta

- Cantidad de cuadros por segundo.
- Resolución.
- Óptica.
- Conexión.
- Iluminación.
- Cantidad de conexiones

Trabajo a futuro

- Flash 11 incluirá un motor de renderizado 3D nativo.
- Estudio del recorrido.
- Un sistema que aprenda.

¡Muchas Gracias!