

Capítulo 3: Método propuesto: Sistema Distribuido de Realidad Aumentada

En este capítulo se describe el desarrollo del framework de Realidad Aumentada distribuido que se propone en este proyecto de tesis, en el que se presenta la arquitectura general con la que se basó la implementación del framework; además de que se presentan las técnicas que fueron utilizadas para la detección de marcadores y la superposición de elementos sobre la escena del mundo real. De igual manera se da una descripción sobre las herramientas y plataformas en las que se implementan en sistemas propuesto.

3.1 Planteamiento.

La propuesta que se presenta en este proyecto de tesis es la del desarrollo de un framework de Realidad Aumentada distribuido; el cual tiene como propósito el mejorar el desempeño en las aplicaciones de Realidad Aumentada en dispositivos móviles, delegando aquellos procesos que demanden de un mayor poder computacional a dispositivos que sean capaces de manejarlos y dejando en este caso a los dispositivos móviles gestionar una cantidad mínima de procesos .

Los puntos principales que se trabajaron en esta tesis, son la disminución de recursos y procesos utilizados en un dispositivo móvil al utilizar este tipo de aplicaciones; además de que se busca que el desarrollo de un sistema de Realidad Aumentada sea más sencillo de implementar, diversificándose sin tantos problemas en diferentes plataformas; ya que como se presentó en un principio la implementación de Realidad Aumentada en dispositivos móviles presenta varias limitaciones y desafíos cuando el sistema es manejado de manera centralizada.

La manera en la que se obtuvieron los resultados mencionados previamente; es a través del desarrollo de un sistema distribuido, delegando en su mayor parte los procesos que requieren de un alto poder computacional a un servidor; sin embargo, una de las mayores limitantes que se pueden presentar ante esta propuesta es en el aspecto de comunicación, ya que mucho dependerá del tiempo de transmisión que se tenga; puesto que a pesar de que exista una disminución de los recursos y procesos utilizados en los dispositivos móviles; si existe un retraso de varios segundos; este tipo de implementación no será adecuada para un sistema de Realidad Aumentada.

3.2 Funcionalidades.

En base a la investigación realizada y al conceptualizar el diseño de la propuesta que se plantea ante el desarrollo de este framework distribuido, se definen los procesos que se ejecutarán en el sistema están determinados por las siguientes acciones:

- Captura de la escena del mundo real: Como se ha mencionado con anterioridad, la principal acción que debe realizar un sistema de Realidad Aumentada, es la extracción de información del mundo físico. Es por ello que para el desarrollo de esta acción se hace solo uso de la cámara presente en los dispositivos móviles. Dado el hecho que no todos los dispositivos de este tipo pueden contar con diversos tipos de sensores, como el GPS, acelerómetros y giroscopios, solo se tomará en cuenta la cámara para realizar la captura del fragmento de la escena real; de tal manera que para el registro y posicionamiento se utilizarán otro tipo de técnicas en base al uso de software.
- Transmisión de la captura del mundo real: Este proceso será llevado a través del uso de servicios web tipo RESTful, en el cual mantendrá la comunicación entre el cliente que en este caso es un dispositivo móvil, el cual es el productor de la imagen

del mundo real; y el servidor en el cual se encuentra el sistema de Realidad Aumentada.

- Obtención de la captura de datos: Al obtener la escena del mundo real a través de los servicios web, la imagen será procesada con el uso del sistema de Realidad Aumentada, de esta manera se observa que la distribución de procesos se está delegando en mayor parte al servidor, dejando que el cliente que son los dispositivos móviles, solo desempeñen las tareas simples de capturar la información del exterior, realizar el envío de los datos capturados, para que posteriormente se obtenga la información procesada por el sistema y está al final pueda ser desplegada en la pantalla del dispositivo móvil.
- Sistema de Realidad Aumentada: Una vez que se haya obtenido la captura de la escena real, el sistema de Realidad Aumentada realizará los procesos de detección de objetos, rastreo y posicionamiento de los elementos virtuales sobre la escena que se capturó; además de la combinación entre lo virtual con lo real. Asimismo durante el proceso de superposición, la extracción de los elementos virtuales se realizará a través de una petición hacia una base de datos o hacia la dirección en donde se encuentren almacenados los elementos de manera local.
- Obtención de la imagen procesada: Una vez que el sistema de Realidad Aumentada haya terminado el proceso de superposición de los elementos virtuales sobre la escena real; el resultado se dispone a ser transmitido hacia el cliente; el cual mantiene una petición constante hacia el servidor para poder obtener la nueva imagen de la escena que se capturó en un principio, teniendo ahora los elementos virtuales incluidos.

- Visualización de la imagen de Realidad Aumentada: Finalmente cuando el cliente obtenga la imagen procesada, esta será desplegada en la pantalla del dispositivo móvil, de tal manera que se tenga la ilusión de que todo el proceso fue gestionado directo en el dispositivo productor.

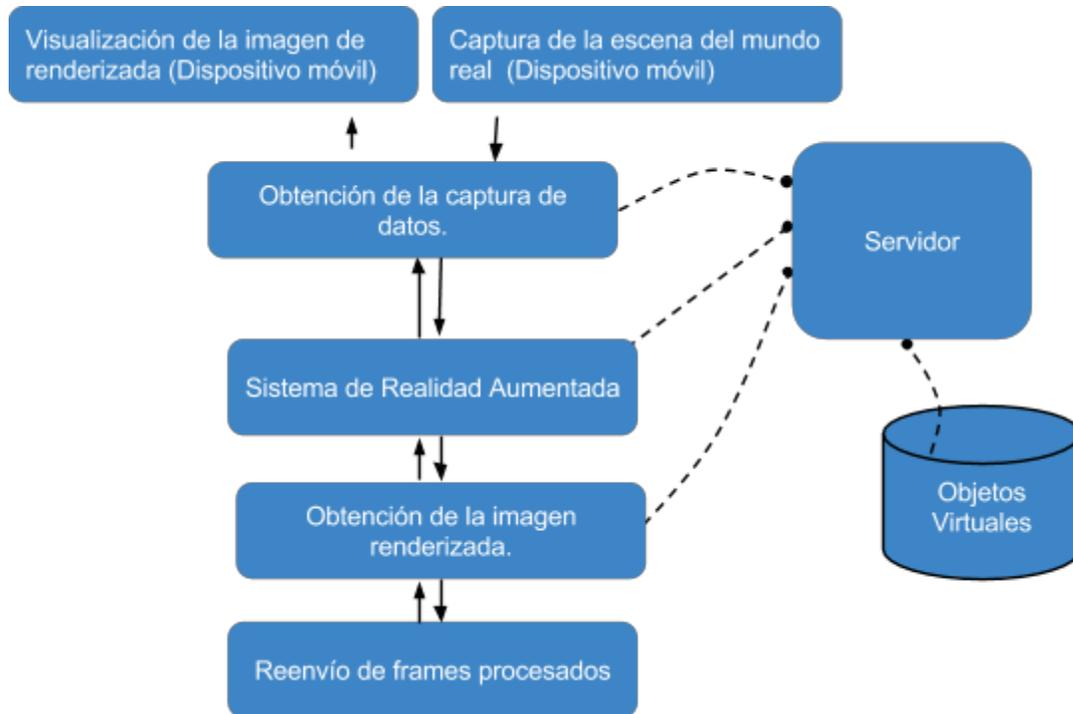


Figura 3.1 Funcionamiento general del framework de Realidad Aumentada distribuido.

3.3 Arquitectura del sistema.

Se optó por hacer uso de una arquitectura cliente/servidor en tres tercios; sin embargo a diferencia de varias implementaciones que fueron descritas en los capítulos anteriores, no se hará uso de cómputo en la nube; y de igual manera en cuanto a distribución de los procesos, no sólo las tareas de rastreo, posicionamiento y detección de marcadores serán realizadas; sino que de igual manera los procesos de obtención y superposición de los elementos virtuales sobre la escena real serán ejecutadas dentro del servidor.

La descripción detallada sobre la distribución de procesos que se realizará en el sistema propuesto distribuido está denotado por los siguientes componentes:

- Cliente: El tipo de cliente que se estará manejando es un cliente ligero; ya que utiliza la mínima cantidad de procesos involucrados en el desarrollo de Realidad Aumentada; de tal manera que las tareas que son gestionadas por el cliente solo es la captura de la escena del mundo real a través de una cámara, que es el dispositivo de entrada que comúnmente se encuentra en cualquier dispositivo móvil; asimismo otro proceso que gestiona el cliente, es el despliegue de información sobre la pantalla.
- Servidor de aplicaciones: El servidor, como se ha mencionado previamente es el componente de la arquitectura, el cual tendrá que realizar la gestión de varios procesos. Entre los procesos que tendrá que ejecutar incluye en su mayoría el desarrollo de la detección de marcadores, rastreo y posicionamiento, además de la superposición de los elementos virtuales. Por otro lado al ser un servidor de aplicaciones, no solo contará con el sistema de Realidad Aumentada; sino que de igual manera, otra aplicación con la que contará será un aplicación web que despliega la escena renderizada, esto con el fin de permitir una depuración del proceso de superposición y de igual manera dentro de esta misma aplicación se podrá realizar la parte de cargar los elementos virtuales y los marcadores a la base de datos.
- Base de datos: La base de datos, solo se encargará de almacenar los elementos virtuales al igual que los identificadores que serán utilizados, de tal manera que se pueda obtener la relación entre el marcador y el objeto virtual al cual está asociado.

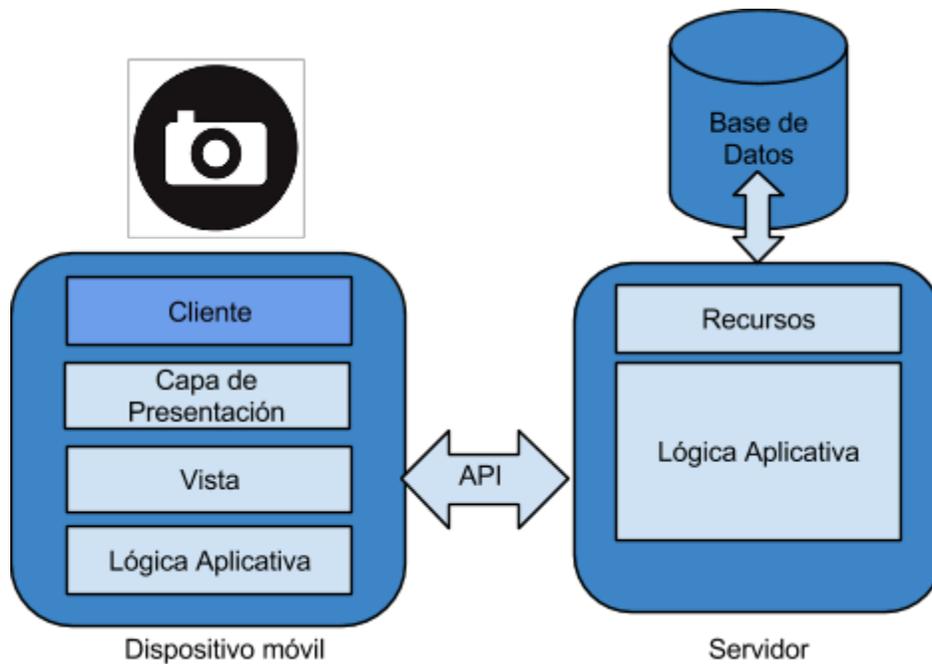


Figura 3.2 Arquitectura del sistema distribuido de Realidad Aumentada.

Por otra parte para poder realizar la comunicación entre los componentes que están presentes en este sistema, se hace uso de los servicios REST. Se optó por el uso de este tipo de servicios web debido a características que presenta como la escalabilidad de componentes, un alto desempeño y eficiencia en la red; además de que proporciona confiabilidad al tener un alto nivel de resistencia a fallas y de igual manera por la simplicidad de las interfaces; es decir, a manera que los componentes comparten información es simple (Pautasso, 2009).

3.4 RESTful API.

Se realizó un RESTful API, el cual describe las diferentes operaciones que son utilizadas durante la comunicación que existe entre el cliente y servidor, o que de igual manera representa la notación de los recursos que provee el servicio web que se está utilizando; los cuales a su vez son representados por URIs.

Las URIs son utilizadas tanto para las peticiones que se realicen con los dispositivos móviles para la implementación del sistema de Realidad Aumentada, como de igual forma para la aplicación web que tiene las tareas de depurar, visualizar y cargar el contenido a la base de datos.

Tabla 1. RESTful API. Recurso: Denota el identificador del recurso al cual se tendrá interacción sobre la red. **Método:** Describe los métodos estándar presentados por el protocolo de comunicación HTTP. **Parámetros:** Se refieren a los datos que deben de proveerse al servicio web, los cuales serán utilizados como información para el desarrollo de nuevo contenido. **Descripción:** Una breve descripción sobre el funcionamiento de cada recurso con su específico método.

Recurso	Método	Parámetros	Descripción
/main	GET	-----	Muestra la imagen que es procesada dentro del servidor, de tal manera que se realice una depuración del proceso de superposición.
/ar2d	GET	-----	Se obtiene un byte array, el cual es la escena del mundo real procesada con el elemento virtual.
	POST	<ul style="list-style-type: none"> ● ByteArray de la imagen capturada. ● Tipo (video o imagen) 	Genera la combinación entre la escena real con los elementos virtuales.
/upload	GET	-----	Muestra los elementos almacenados en la base de datos.
	POST	<ul style="list-style-type: none"> ● ByteArray de marcador y del elemento virtual. 	Guarda el elemento virtual y su identificador en la base de datos.

3.5 Diagramas.

A continuación se muestran los diagramas de caso de usos y de clases de sistema de Realidad Aumentada, con la finalidad de presentar de manera más detallada el diseño del sistema.

3.5.1 Diagrama de caso de usos.

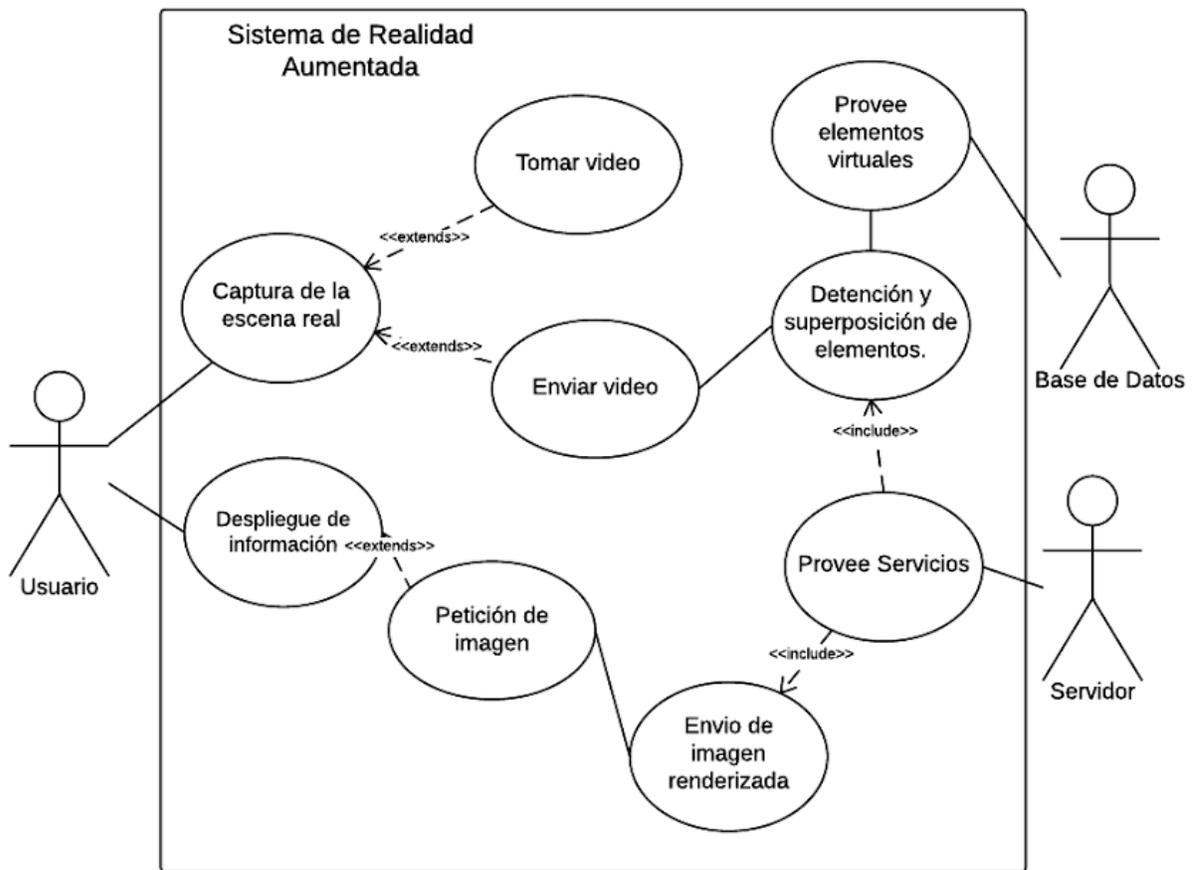


Figura 3.3. Diagrama de casos de usos, el cual describe el comportamiento del sistema.

3.5.2 Diagrama de clases.

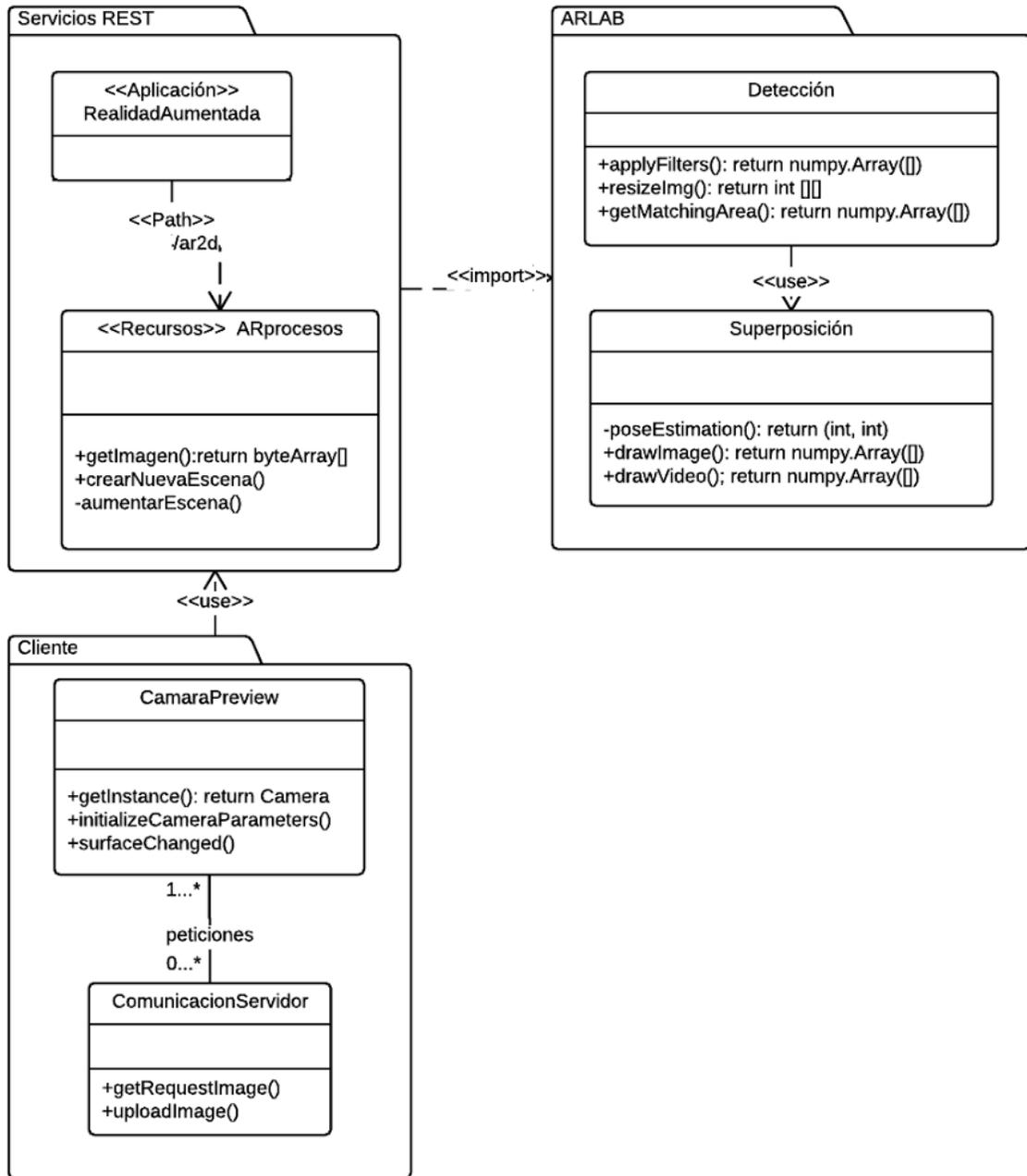


Figura 3.4. Diagrama de clases del framework de Realidad Aumentada, el cual describe la estructura general del sistema, de igual manera se observa el diseño en cuanto a la distribución de procesos.

3.6 Componentes y herramientas de desarrollo.

3.6.1 Componentes físicos.

Los componentes que se utilizaron en el desarrollo del framework de Realidad Aumentada que se propone en este proyecto de tesis son:

- Dispositivos móviles: Se utilizaron diferentes tipos de dispositivos móviles como lo fue una laptop, una tableta y un teléfono inteligente, los tres dispositivos presentan diferentes especificaciones.
- Cámara: Como dispositivo de entrada para realizar la captura de las escenas del mundo real se utilizó la cámara de los dispositivos móviles; no se tomó en cuenta el uso de sensores puesto que no todos los dispositivos móviles cuentan con ellos.
- Pantalla: Se utilizó la pantalla presente en los dispositivos móviles como medio de salida para desplegar la información que se está generando.
- Servidor: Para la parte del servidor se utilizó una laptop, que cumple con las expectativas para poder implementar los procesos que requiere el sistema de Realidad Aumentada.
- Red Wireless: La red wireless permite establecer la comunicación entre los componentes que están presentes en este sistema.

3.6.2 Herramientas de desarrollo.

Asimismo las herramientas que se utilizaron en el desarrollo del software que está presente en el sistema de Realidad Aumentada son:

- Python: Se utilizó el lenguaje de programación python para el desarrollo de los servicios REST, además de las tareas del sistema de Realidad Aumentada que se ejecutan en el servidor; por otra parte la aplicación web destinada a la depuración y a la subida de archivos a la base de datos igualmente se desarrollo en python. Se optó por el uso de este lenguaje debido a que el prototipado de un programa puede desarrollarse de manera más simple y rápida; inclusive para el desarrollo web.
- OpenCV: Es una librería utilizada para el desarrollo de aplicaciones relacionadas con el campo de Visión por Computadora, que en este caso es utilizada para las tareas que se llevan a cabo en el sistema de Realidad Aumentada, tales como la detección de marcadores y la superposición de los elementos virtuales sobre la escena real.
- SQLite: Debido a que solo se realizaran pruebas que no requieren del almacenamiento de una gran cantidad de contenido, en este caso de los elementos virtuales; se optó solo por usar SQLite para gestionar el almacenamiento de los elementos virtuales y sus identificadores.
- Android SDK y JAVA: El hecho de que dos de los dispositivos móviles que se están utilizando para realizar las pruebas del sistema de Realidad Aumentada usan el sistema operativo Android; es necesario contar con las herramientas necesarias para la compilación y ejecución del código.

3.7 Detección de marcadores.

La detección de marcadores resulta crucial, para un sistema de Realidad Aumentada (Biocca & Levy, 2013), de tal manera que con ello se puede determinar el lugar en el que los elementos virtuales serán superpuestos. En este caso se utiliza la librería de OpenCV, la cual provee varios algoritmos para resolver problemas presentes en Visión por Computadora; lo cual es de gran utilidad ya que se requiere de la detección de objetos sobre una determinada escena.

En cuanto a los marcadores o identificadores que serán registrados para determinar el posicionamiento de los elementos virtuales, se utilizarán los frame markers, los cuales son más fáciles de identificar en la escena real, además de que el proceso de detección para este tipo de marcadores no requiere de un alto poder computacional.

Por otra lado, el proceso de detección de marcadores utilizando en el sistema de Realidad Aumentada, está conformado por subprocesos tales como:

- Cargar en el sistema la plantilla que es utilizada para identificar el marcador; además de la escena en la que se buscará el marcador.
- Posteriormente se hace uso de la función proporcionada por OpenCV, llamada *matchTemplate*, en esta función se utiliza uno de los métodos más comunes cuando se realizan comparaciones de imágenes. El método utilizado es el de correlación cruzada normalizada; este método consiste en dividir en regiones a la imagen que se está analizando, de tal manera que se pueda realizar la comparación con la plantilla o marcador que se desea detectar; de igual forma se optó por el uso de este método debido a la precisión considerable que tiene para detectar objetos al igual que por su simplicidad. De manera más específica se hace un análisis de cada pixel de la imagen, de izquierda a derecha, de la parte superior a la inferior de la imagen, de tal

manera que se calcule la distancia entre dos matrices, la cual determina que tan parecido es una imagen con respecto a otra, en este caso lo que se busca es obtener la región de la escena real que tenga más parecido a la plantilla que se utiliza.

- Finalmente el resultado que fue calculado es normalizado, con el fin de poder obtener el punto de intensidad más alto; el cual determina el lugar dentro de la imagen analizada en la que se encuentra la plantilla deseada. El método de correlación cruzada está denotado por la siguiente fórmula dada por Mordvintsev y Rahman:

$$R(x, y) = \frac{\sum_{x', y'} (T(x', y') * I(x+x', y+y'))}{\sqrt{\sum_{x', y'} (T(x', y')^2 * \sum_{x', y'} I(x+x', y+y')^2)}}$$

- Una vez que se ha localizado el área donde el elemento virtual será superpuesto, se puede determinar la región de la imagen en la que será añadido el objeto virtual.

3.8 Superposición de objetos virtuales.

Finalmente el último proceso que realiza cualquier sistema de Realidad Aumentada y el cual a su vez es el principal proceso que no puede ser omitido en el desarrollo de estos sistemas; es el de la integración de los elementos virtuales sobre la escena del mundo real.

Para ello existen una gran variedad de técnicas, especialmente dependiendo del tipo de elementos virtuales con los que se estén trabajando; en este caso solo se hará uso de elementos en 2D, por lo que se identificó una función proporcionada por OpenCV, ante el concepto de ROI (Brinkmann ,1999); puesto que ROI hace referencia hacia una región de interés de una imagen, se puede utilizar de tal manera que al aplicar una simple operación

bitwise AND, se pueda extraer una área específica de la imagen, la cual es combinada con el elemento virtual que se desea superponer; teniendo como resultado que la combinación entre el elemento virtual con la escena del mundo real, pueda ser generada.

Una vez que se haya completado el proceso de superposición, se genera una nueva imagen, la cual será transmitida como resultado del sistema de Realidad Aumentada; esta nueva imagen será consumida por los dispositivos móviles la cual es visualizada en la pantalla de dichos dispositivos; generando de esta manera la ilusión que todo el proceso se está ejecutando de manera centralizada y evitando de igual manera las limitantes que están presentes en los dispositivos móviles para el desarrollo de sistemas de Realidad Aumentada.

3.9 Resumen.

En este capítulo se dio una introducción a fondo sobre la arquitectura y diseño presentes en el desarrollo del framework de Realidad Aumentada; en el que se presentaron los componentes y herramientas que se utilizaron, dando una explicación sobre el uso que desempeñan en el sistema que se desarrolló, además de una justificación de su elección.

Asimismo se habla sobre los procesos más relevantes dentro del sistema de Realidad Aumentada como lo es la detección de marcadores y la superposición de los elementos virtuales sobre la escena real.