

Incorporación de la tecnología de realidad aumentada a un vehículo aéreo no tripulado

Incorporation of augmented reality technology to an unmanned aerial vehicle

Alex Feng¹; Andrés García¹; Kevin Torres¹ & Euclides Samaniego González^{2*}

¹Licenciatura en Ingeniería de Sistemas y Computación – Facultad de Ingeniería de Sistemas Computacionales
– Universidad Tecnológica de Panamá

²Grupo de Investigación en Inteligencia Computacional – GIICOM
– Facultad de Ingeniería de Sistemas Computacionales – Universidad Tecnológica de Panamá

Resumen Este documento es un informe del diseño, ensamblaje, pruebas y resultados obtenidos, de la construcción de un vehículo aéreo no tripulado con un sistema de realidad aumentada incorporado para la visualización de modelos digitales desde una perspectiva aérea.

Palabras claves Detección de marcadores, realidad aumentada, vehículo aéreo no tripulado, RPM, *frame*, UAV.

Abstract This document serves as a report about the design, assembly, testing a results obtained from the construction of an unmanned aerial vehicle which has an augmented reality system incorporated to allow the showing of digital models from an aerial perspective.

Keywords Markers detection, augmented reality, unmanned aerial vehicle, RPM, frame, UAV.

* **Corresponding author:** euclides.samaniego@utp.ac.pa.

1. Introducción

Este proyecto surgió al observar el crecimiento que ha tenido en los años recientes el uso de vehículos aéreos no tripulados en nuestro país, al investigar, se percató de que al igual que muchos otros productos electrónicos, no son producidos en nuestro país sino importados, como es el caso de sus piezas y refacciones.

Por lo anteriormente expuesto, se decidió construir uno de acuerdo a las necesidades, con las piezas disponibles de tal forma no verse limitados por los modelos existentes en el mercado, de manera análoga, se había planteado la posibilidad de realizar un proyecto empleando realidad aumentada, por lo que se decidió integrar ambos proyectos y producir un vehículo aéreo no tripulado en incorporar la funcionalidad de observar modelos de edificaciones generados por computadora a través de realidad aumentada.

Permitiendo así analizar la posición, tamaño e impacto de dichas estructuras a su entorno incluso antes de la construcción de las mismas.

Al utilizar un dron incorporamos la posibilidad de observar los modelos desde una perspectiva aérea, que en comparación con otras ideas como usar helicópteros o plataformas, posee un bajo costo además de una mayor maniobrabilidad en ambientes reducidos o con obstáculos.

Los objetivos de este estudio son:

- Entender los conceptos básicos para el desarrollo de drones y realidad aumentada.
- Diseñar un vehículo aéreo no tripulado con un sistema de video y transmisión de imágenes.
- Aplicar la detección de marcadores a las imágenes captadas.
- Mostrar modelos 3D sobre los marcadores detectados en las imágenes captadas.

En ese artículo exponemos el diseño del dron, junto con el de la realidad aumentada, describimos el proceso de construcción del vehículo y desarrollo de la aplicación, junto con descripciones de las pruebas realizadas y finalmente los resultados obtenidos de nuestros experimentos.

1.1 Vehículo aéreo no tripulado

Se entiende por VANT (Vehículo Aéreo No Tripulado) o UAV (Unmanned Aerial Vehicle) a aquella nave que es capaz de realizar una misión sin necesidad de tener una tripulación embarcada. Debe entenderse que esta condición no excluye la existencia de piloto, controlador de la misión u otros operadores que pueden realizar su trabajo desde tierra [1].

Desde sus inicios, los drones eran utilizados exclusivamente para aplicaciones militares, las cuales han ido desde el entrenamiento de soldados para acertar a objetivos en movimiento hasta para la captura de imágenes de reconocimiento en tiempo real.

En años recientes, su uso se ha extendido desde el simple uso militar a otros sectores, como el científico, comercial y recreacional.

Los drones han permitido realizar tareas que antes eran difíciles o riesgosas si fueran realizadas por un ser humano, ya sea por la dificultad de llegar al lugar, o por el peligro que involucraría la tarea en sí.

Además de su utilización por parte de instituciones científicas, gubernamentales y empresas, su uso ha llegado a personas civiles para uso recreacional, mayormente se emplean para tomar videos aficionados o jugar, su facilidad de uso ha llegado al punto donde incluso niños de 6 años pueden utilizar estos dispositivos.

1.2 Realidad aumentada

Realidad aumentada es aquella tecnología que permite al usuario ver en todo momento el mundo real, al que se le superponen objetos virtuales coexistiendo ambos en el mismo espacio [2].

Se puede decir que la realidad aumentada combina elementos virtuales generados por computador con elementos del entorno físico, creando una vista entre lo real y lo virtual.

Durante la década de 1990, las aplicaciones de realidad aumentada dentro del ámbito industrial y el militar continuaron desarrollándose. Sin embargo, el alto costo y nivel de complejidad de los requerimientos técnicos para su desarrollo

mantuvieron estas tecnologías lejos del alcance de la mayoría de los usuarios.

Actualmente la realidad aumentada es aplicada en diferentes áreas debido a las posibilidades de interacción que ofrece, permitiendo que se aplique en varios ámbitos, como lo es la arquitectura, donde es empleada para visualizar los proyectos de construcción, ya sea en la mesa de diseño o en el lugar donde se construirá finalmente, además de permitir reconstruir virtualmente monumentos históricos destruidos [3].

Entre las áreas más explotadas de la realidad aumentada está la educación, se emplea en museos, exhibiciones y parques temáticos. En estos lugares se utilizan conexiones inalámbricas para transmitir y mostrar información sobre objetos y lugares o imágenes virtuales como ejemplos de ruinas reconstruidas o paisajes tal y como eran en el pasado [4].

No solo se aplica en la historia, sino también en clases como química, donde se presenta la estructura de las moléculas, o en biología, donde permite observar los órganos del cuerpo humano observar su funcionamiento.

2. Diseño del sistema

2.1 Requerimientos del vehículo aéreo no tripulado

Debido a la gran cantidad de componentes con distintas características que se pueden adquirir en el mercado, es imprescindible un análisis de requerimientos para determinar las características que va a poseer el vehículo aéreo no tripulado.

2.1.1 Tamaño del vehículo aéreo no tripulado

El tamaño del vehículo aéreo no tripulado va ligado al tamaño del armazón o *frame*. El armazón debe ser de un material rígido y a la vez ligero, por eso se escogió uno de fibra de carbono con un centro plano para colocar los componentes electrónicos.

Este *frame* tiene una distancia de motor a motor de 48 cm con una configuración en X para permitir la instalación de los otros componentes.

Este nos permite incorporar hélices de

hasta 30.48 cm o 12 plg, sin embargo, hemos decidido emplear hélices de 25,4 cm o 10 plgs de diámetro ya que nos brinda una separación entre hélices lo suficiente para asegurar que no se golpeen y para que el aire fluya de manera correcta, evitando la turbulencia que podría generar la fuerza del aire de cada hélice.

2.1.2 Fuerza de empuje de los motores

Para la elección de los motores de un vehículo aéreo no tripulado hay que seguir cierto criterio que se muestra a continuación:

- Cuando se utilizan motores con alto RPM ($k_v > 1000$), se debe utilizar hélices de tamaño pequeño o mediano.
- Cuando se utilizan motores con bajo RPM ($k_v < 1000$), se debe utilizar hélices de tamaño grande. Si se utilizan hélices de tamaño pequeño, probablemente no sea capaz de levantar el vehículo aéreo no tripulado.

Se requieren motores que sean eficientes en el consumo de corriente. Para motores con 935 kv y una fuente de alimentación de 12V, la cantidad de revoluciones por minutos (RPM) son:

$$\begin{aligned} \text{RPM} &= k_v * U & (1) \\ \text{RPM} &= 935 k_v (12V) \\ \text{RPM} &= 11\ 220 \text{ RPM} \end{aligned}$$

Esta cantidad de revoluciones por minuto es ideal para vehículos aéreos no tripulados que requieren estabilidad y levante de peso.

Otro de los factores a tomar en cuenta de los motores es la fuerza de empuje de cada motor. Cada motor individual debe tener una fuerza de empuje de 850 gramos, así, al combinar los cuatro, obtendríamos una fuerza de empuje teórica:

$$\begin{aligned} F_{Et} &= F * 4 & (2) \\ F_{Et} &= 850 \text{ g} * 4 \\ F_{Et} &= 3\ 400 \text{ gramos} = 3.4 \text{ kg} \end{aligned}$$

2.1.3 Fuente de energía y tiempo de vuelo

Para obtener un aproximado de la potencia que debe tener la fuente de energía y el tiempo

de vuelo del vehículo aéreo no tripulado, debemos realizar algunos cálculos:

En base a los datos que muestra la tabla 1, la corriente consumida por un motor EMAX MT2213 para una fuerza de empuje de 290 g es de 3A.

Tabla 1. Consumo de corriente según el tamaño de hélice y fuerza de empuje de un motor EMAXMT2213 935kv

The voltage (V)	Paddle size	current (A)	thrust (g)	power (W)	efficiency (G/W)	speed (RPM)	Working temperature (°C)
11	EMAX8045	1	110	11	10.0	3650	
		2	200	22	9.1	4740	
		3	270	33	8.2	5540	
		4	330	44	7.5	6200	
		5	390	55	7.1	6700	
		6	440	66	6.7	7150	
		7.1	490	78.1	6.3	7400	36
	EMAX1045	1	130	11	11.8	2940	
		2	220	22	10.0	3860	
		3	290	33	8.8	4400	
		4	370	44	8.4	4940	
		5	430	55	7.8	5340	
		6	480	66	7.3	5720	
		7	540	77	7.0	5980	
		8	590	88	6.7	6170	
		9	640	99	6.5	6410	
		9.6	670	106	6.3	6530	43

Como son 4 motores, entonces:

Consumo = consume por motor * cantidad de motores

$$\begin{aligned} \text{Consumo} &= 3A * 4 \\ \text{Consumo} &= 12A \end{aligned} \quad (3)$$

Si la corriente que se consume para poder flotar son 12 amperios (x), el tiempo que queremos de vuelo es T (en horas). Como el tiempo de vuelo que queremos son de 10 minutos, entonces la capacidad (C) es:

$$\begin{aligned} C &= xT \\ C &= 12A(0,1666 \text{ h}) \\ C &= 2,00Ah = 2000mAh \end{aligned} \quad (4)$$

De estos cálculos tenemos que se necesita una batería de 2000mAh para que el vehículo aéreo no tripulado pueda flotar o sostenerse en el aire.

Este cálculo no es totalmente cierto ya que estamos suponiendo que se utilice toda la carga

de la batería, pero esto no puede suceder ya que las baterías tipo LiPO no se pueden descargar totalmente y tienen un límite mínimo de corriente que puede suplir.

Como las baterías LiPO nunca deben descargarse del todo debemos considerar la carga aprovechable. En este caso, utilizamos el 50% de la carga de la batería, entonces tenemos que la capacidad real de la batería sería:

$$\begin{aligned} C' &= \frac{C}{(\text{carga aprovechable})} \quad (5) \\ C' &= \frac{2000mAh}{0.50} \\ C' &= 4000 \text{ mAh} \end{aligned}$$

Por lo tanto, la fuente de energía del vehículo aéreo no tripulado con motores EMAX MT2213 935kv con hélices de 10 pulgadas de diámetro y 4,5 pulgadas de pitch, debe ser de 4000 mAh con una carga aprovechable de al menos 50%, lo cual nos proporciona un tiempo de vuelo alrededor de 10 minutos.

2.2 Esquema conceptual del vehículo aéreo no tripulado

En la figura 1 se muestra el esquema conceptual del vehículo aéreo no tripulado.

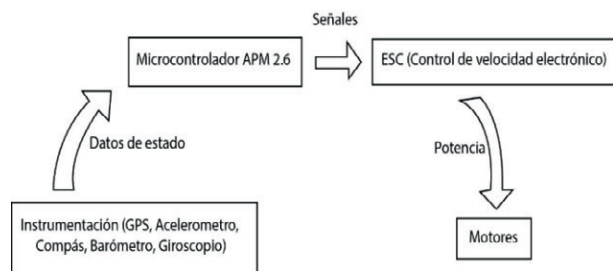


Figura 1. Esquema conceptual del prototipo.

2.3 Diseño estructural del prototipo

En la figura 2 se muestra el diseño estructural del vehículo aéreo no tripulado empleado en el desarrollo del proyecto.

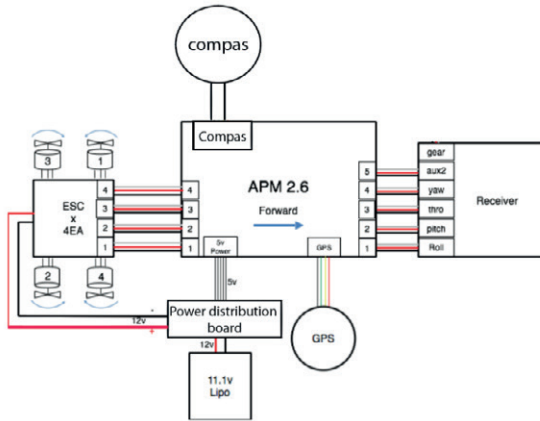


Figura 2. Diseño estructural del prototipo de vehículo aéreo no tripulado.

2.3.1 Elementos a incorporar

APM (*ArduPilot Mega Flight Controller Board*): ArduPilot es un proyecto *open source* basado en el sistema Arduino.

El mismo consiste en una plataforma que conforma una unidad de control de navegación, la cual puede ser utilizada en diferentes proyectos como en aeronaves, coches y vehículos aéreos no tripulados de despegue vertical. Ver figura 3.



Figura 3. Andoer APM 2.6 Flight Controller Board.

Motor eléctrico sin escobillas (Brushless motors): en la figura 4 se presenta el motor eléctrico sin escobillas Motores *brushless* Emax Mt2213 935kv.



Figura 4. Motores *brushless* Emax Mt2213 935kv.

ESC (control eléctrico de velocidad): debido a que estamos utilizando un motor sin brochas, es necesaria la implementación de un controlador eléctrico de velocidad. Ver figura 5.

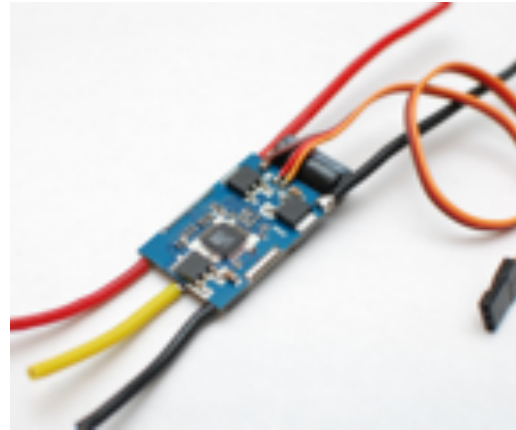


Figura 5. Un módulo genérico de salida nominal de 35 amperios.

Batería de polímero de litio: una batería de polímero de litio, o más correctamente la batería de polímero de iones de litio (abreviado vario como LiPo, labio, Li-poly y otros), es una batería recargable de la tecnología de iones de litio en forma de bolsa.

Cargador para baterías LiPo: las baterías tipo LiPo requieren de cargadores especiales para sus celdas ya que estas están separadas y cada una tiene un voltaje independiente. Ver figura 6.



Figura 6. Cargador Imax B6AC.

Control de radio con receptor; el control del vehículo aéreo no tripulado desde tierra se da a través de un radio control el cual, como su nombre lo indica, utiliza señales de radio para transmitir las órdenes. Ver figura 7.



Figura 7. Flysky FS-i6 2.4GHz.

Tablero de distribución de energía: un tablero de distribución (también conocido como tablero o interruptor de panel) es un componente de un sistema de suministro de electricidad que se divide una alimentación de energía eléctrica en circuitos auxiliares, mientras que proporciona un fusible o un disyuntor para cada circuito en una caja común (ver figura 8).



Figura 8. Tablero de distribución de carga.

Módulo GPS: el sistema de posicionamiento global (GPS) es un sistema que permite determinar en todo el mundo la posición de un objeto (una persona, un vehículo). Ver figura 9.



Figura 9. Módulo GPS.

Transmisor de audio y video inalámbrico; un transmisor es un dispositivo electrónico que, con la ayuda de una antena, irradia ondas electromagnéticas que contienen información y que encuentra aplicaciones en radiodifusión, televisión, telefonía móvil entre otros.

En la figura 10 se presenta una imagen del modelo de transmisor de audio y video inalámbrico utilizado en el prototipo construido.



Figura 10. Gooray 32Ch 5.8G 600mw 5km Wireless AV transmisor TS832 Receptor RC832 para FPV.

Convertidor de señal AV analógica a digital: la señal recibida es de tipo analógica.

Como se va a procesar la imagen por computador se necesita una señal digital.

Este componente se encarga de realizar la transformación de señal. Ver figura 11.



Figura 11. EasyCAP USB Video Capture.

Cámara CMOS: gracias a la tecnología CMOS es posible integrar más funciones en un chip sensor, como por ejemplo control de luminosidad, corrector de contraste, o un conversor analógico-digital. Ver figura 12.



Figura 12. Cámara CMOS FOV160 FPV200.

2.3.2 Ensamblado de los componentes del VANT

Como primera parte, se prepara el armazón o estructura que funcionara como base y contenedor de todos los componentes del vehículo aéreo no tripulado. En la figura 13 se muestra la estructura ya construida y ensamblada.



Figura 13. Armazón o estructura tipo X.

Una vez que el armazón estuvo ensamblado, atornillamos los motores al armazón y conectamos los ESC a cada uno de los motores.

No todas las hélices pueden girar en un mismo sentido, por ello que se intercalan la rotación de los motores, es decir, el primer rotor gira en sentido de las manecillas del reloj; el segundo rotor gira en contra de las manecillas del reloj y así, sucesivamente.

La conexión de los ESC al microcontrolador APM 2.6 se muestra en la figura 14.

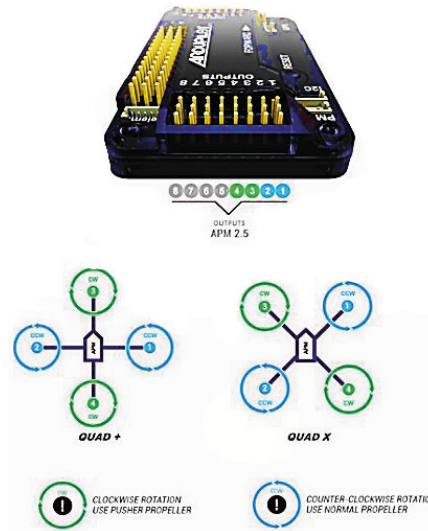


Figura 14. Rotación de los motores de acuerdo a su disposición en la estructura y pines correspondientes en el APM 2.6.

Es importante seguir la enumeración ya que, en el momento de la configuración del microcontrolador, se supone que los ESC de los motores están dispuestos de esa manera.

Antes de conectar los motores, hay que calibrar los ESC ya que estos componentes vienen calibrados con otra configuración de fábrica.

Al calibrarlo nos aseguramos que el ESC va a responder correctamente a las señales que le enviemos por medio del microcontrolador y establecemos los puntos máximos y mínimos de potencia. Para calibrarlo se debe seguir el esquema de la figura 15.

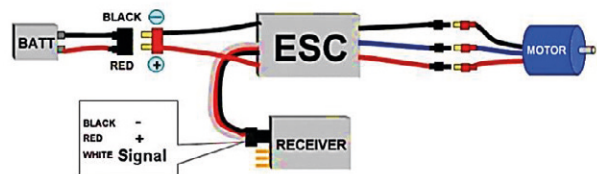


Figura 15. Esquema para calibrar ESC.

En la calibración, el ESC no debe estar conectado a la batería y en el mando de radio control el *throttle* o elevación debe estar al máximo.

Una vez el *throttle* o elevación estén al máximo, se conecta la batería y se deben escuchar unos sonidos como “beeps” que son tonos de información.

Justo después de escuchar los tonos de información se debe bajar al mínimo el *throttle* o elevación hasta escuchar otros “beeps”.

La alimentación de los motores la supe la batería que va conectada a una placa de distribución de energía como el de la figura 16.

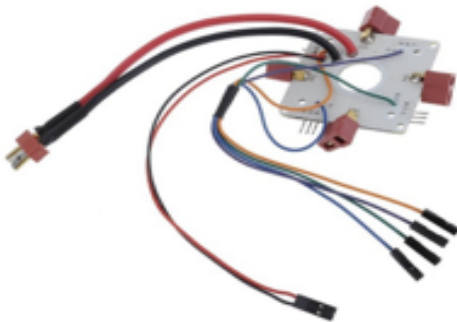


Figura 16. Placa de distribución de energía.

De esta forma nos aseguramos que la misma intensidad de corriente y voltaje sea igual para todos los motores.

Para comunicar el radio control con el microcontrolador se utiliza un receptor de señales de radio.

El receptor de señales de radio que se muestra en la figura 17, consta de seis canales de los cuales solo se utilizan cinco de ellos.



Figura 17. Receptor de señal FS-IA6.

La descripción de los canales (ver figura 18) que se utilizaron se muestra a continuación:

- Canal 1. Control del *Roll*
- Canal 2. Control *Pitch*
- Canal 3. Control *Throttle* o elevación
- Canal 4. Control *Yaw*
- Canal 5. Control del modo de vuelo

2.3.3 Ensamblado de los componentes del sistema de video

La cámara de video se debe colocar al frente del UAV para no tener obstáculo que impidan la visualización. Esta cámara se conecta con un emisor de señales de analógicas y los envía de forma inalámbrica con señales de radio.

Todo este conjunto se alimenta de una batería de 12V para su funcionamiento.

En la figura 19 se muestra el emisor de señales analógicas y la cámara de video respectivamente.

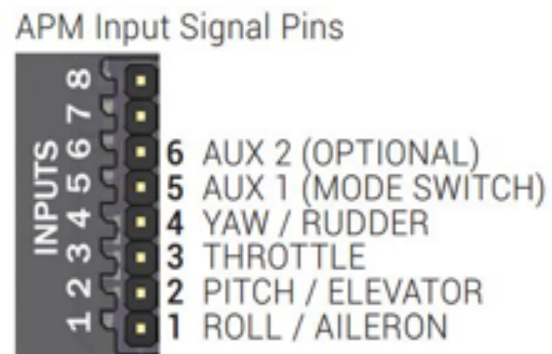


Figura 18. Señales de entrada del APM 2.6.



Figura 19. Emisor de señales analógicas LT200.

2.4 Diseño de la aplicación de realidad aumentada

Para el desarrollo de la aplicación de realidad aumentada se usó Windows como sistema operativo.

Las aplicaciones utilizadas fueron Visual Studio 2013, en este se trabajó en el entorno del lenguaje de programación C++ aplicando la librería de *OpenGL*.

De la librería *OpenGL*, se usó una en específica para el desarrollo de realidad aumentada una variación llamada *artoolkit*.

La aplicación al ejecutarse en el sistema operativo Windows, recibe el video por medio del convertidor de audio video, este lo analiza y trata de detectar el marcador para proyectar cualquiera de los modelos gráficos cargados en el programa.

En este caso se cargaron cinco modelos gráficos que se pueden cambiar al pulsar los teclados numéricos del 1 al 5.

Además, desde el teclado se puede manipular el modelo gráfico, ya se tiene designado las teclas w y s para escalar y las teclas de direccionamiento para trasladar.

El marcador que se utilizó en el programa fue el que estaba cargado en la librería de *artoolkit* el cual tiene una forma rectangular y en su interior tiene la palabra “Hiro”.

En la figura 20 se muestra una imagen del tipo de marcador utilizado.

2.4.1 Diagrama de flujo de datos

En la figura 21 se muestra el diagrama de flujo de datos de una aplicación de realidad aumentada.

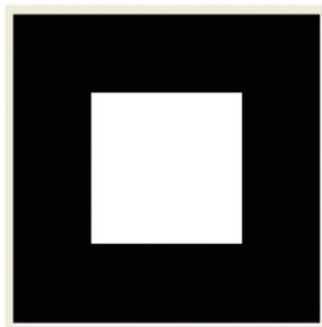


Figura 20. Ejemplo marcador.

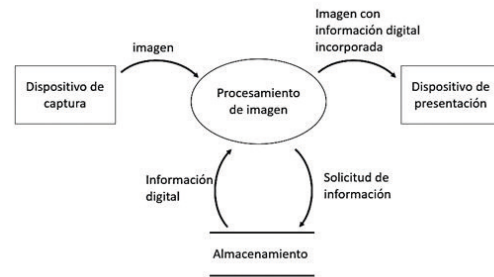


Figura 21. Diagrama de flujo de datos de una aplicación de realidad aumentada.

2.4.2 Programación de los módulos para realidad aumentada

El principal atractivo de la realidad aumentada es la presentación de modelos digitales.

Por lo antes expuesto nuestro programa inicia por la transferencia a memoria de los modelos 3D que se visualizarán, eso se realiza a través la lectura de los archivo con extensión “.obj”, el cual contiene la forma del modelo, y el archivo con extensión “.mtl” el cual contiene los colores del modelo.

La figura 22 muestra el ejemplo de código para cargar el modelo utilizamos la función:

```
loadOBJ(char *path, char *pathMTL)
```

Figura 22. Ejemplo de código.

De la figura anterior se tiene que *path* = dirección del archivo .OBJ
pathMTL = dirección del archivo .MTL

Dentro de la función `loadOBJ(char *path, char *pathMTL)` se evalúa el contenido dentro de cada archivo buscando las coordenadas de los vértices, de las texturas y las normales; como se muestra en la figura 23.

Una vez obtenidos todos los datos, dentro del archivo .OBJ existe el indicador “f” que es un conjunto de 9 valores, 3 para vértices para una cara, 3 valores para las texturas y 3 valores para la normal.

Estos valores se deben organizar y se realiza con el segmento de código que se muestra en la figura 24.

```

fscanf(file, "%s", &ch); /*busca los vertices*/
if (strcmp(ch, "v") == 0)
{
    float x, y, z;
    vec3 vertex;
    fscanf(file, "%f %f %f\n", &vertex.x, &vertex.y,
&vertex.z);
    temp_vertices.push_back(vertex);
}
else if (strcmp(ch, "vt") == 0) /*busca las texturas*/
{
    vec2 uv;
    fscanf(file, "%f %f\n", &uv.x, &uv.y);
    temp_uv.push_back(uv);
}
else if (strcmp(ch, "vn") == 0) /*busca las normales*/
{
    vec3 normal;
    fscanf(file, "%f %f %f\n", &normal.x, &normal.y,
&normal.z);
    temp_normals.push_back(normal);
}

```

Figura 23. Ejemplo de código.

El contenido de ambos archivos es almacenado en vectores para su posterior uso en el programa de manera que mejora el rendimiento del programa al no tener que pasar a memoria el modelo cada vez que el mismo tenga que ser dibujado.

```

out_vertices.push_back(temp_vertices[vertexIndex[0] - 1]);
out_vertices.push_back(temp_vertices[vertexIndex[1] - 1]);
out_vertices.push_back(temp_vertices[vertexIndex[2] - 1]);

triangDef.coord1 = temp_vertices[vertexIndex[0] - 1];
triangDef.coord2 = temp_vertices[vertexIndex[1] - 1];
triangDef.coord3 = temp_vertices[vertexIndex[2] - 1];

triangDef.normal1 = temp_normals[normalIndex[0] - 1];
triangDef.normal2 = temp_normals[normalIndex[1] - 1];
triangDef.normal3 = temp_normals[normalIndex[2] - 1];

```

Figura 24. Ejemplo de código.

2.4.3 Calibración de cámara

Inicialmente se solicitan por lo menos 10 imágenes, sin embargo, se pueden usar más, al terminar la calibración, el programa utiliza un archivo “camera_para.dat”, el cual deberá reemplazar el archivo anterior que viene por defecto en los archivos del programa. Ver figura 25.

```
const char *patt_name = "Data/patt.hiro";
```

Figura 25. Ejemplo de código.

También se utiliza un archivo *xml* que contiene los parámetros generales de distintas cámaras.

Una vez obtenidos los valores de los archivos se realiza la configuración con el segmento de código descrito en la figura 26.

```

if (!setupCamera(cparam_name, vconf, &gARTCparam)) {
    fprintf(stderr, "main(): Unable to set up AR camera.\n");
    system("pause");
    exit(-1);
}

```

Figura 26. Ejemplo de código.

Si la configuración de la cámara es exitosa continúa con los siguientes procesos, en caso contrario, rompe todos los procesos hasta ese punto.

2.4. Detección de marcadores

Para configurar los marcadores, primero debemos cargar el archivo del marcador que se vaya a utilizar en el programa.

```
const char *patt_name = "Data/patt.hiro";
```

Luego se manda el archivo a memoria empleando el segmento de código como se muestra en la figura 27.

```

if (!setupCamera(cparam_name, vconf, &gARTCparam)) {
    fprintf(stderr, "main(): Unable to set up AR camera.\n");
    system("pause");
    exit(-1);
}

```

Figura 27. Ejemplo de código.

Si la configuración del marcador es exitosa continúa con los siguientes procesos, en caso contrario, rompe todos los procesos hasta ese punto.

Una vez configurados todos los componentes para que la librería ARToolkit funcione se procede a captar las imágenes de la cámara y procesar los marcadores detectados para cargar los modelos 3D.

3. Resultados experimentales

A través de este proyecto se logró construir un cuadri-cóptero de especificaciones superiores a las existentes en el mercado, como es en el caso de su alcance de 1000 metros en comparación con los 300 metros de los modelos comerciales actuales.

Además de esto, es capaz de levantar dos libras además de su propio peso, por lo que es capaz de transportar artículos pequeños de un lugar a otro.

El vehículo aéreo no tripulado cuenta con un sistema de video incorporado el cual transmite imágenes en tiempo real a una computadora,

donde un programa de realidad aumentada presenta las imágenes con modelos digitales en 3D incorporados para su visualización.

Resultó en una imagen de poca calidad ya que el medio de transmisión no es el mejor, además que se convierte la señal múltiples veces creando mucho ruido en la imagen final. Ver figura 28.



Figura 28. Imagen obtenida con el sistema de cámara del VANT.

El tiempo de vuelo se midió con un cronómetro el cual marcó un tiempo de vuelo de 15 minutos con la batería cargada al máximo.

En las pruebas de vuelo, tiene una maniobrabilidad media ya que, por el tamaño, permite mayor estabilidad que maniobrabilidad. Ver figura 29.

Esto quiere decir que para escenarios donde exista una gran cantidad de obstáculos y giros bruscos, este tipo de vehículo no es el más apto. Sin embargo, para escenarios donde existan pocos obstáculos y ráfagas de aire fuerte, este vehículo es perfecto para estas condiciones.

Otro resultado que no esperábamos obtener es la resistencia a golpes y caídas. Durante las pruebas de vuelo, el vehículo sufrió múltiples caídas, de estas podemos destacar una caída equivalente a cuatro pisos de edificios.

El vehículo resultó con golpes y algunos soportes de motores se quebraron, pero todo el funcionamiento electrónico quedó intacto.

4. Resultados y conclusiones

En este proyecto utilizamos los vehículos aéreos no tripulados para incorporarle la tecnología de realidad aumentada como forma



Figura 29. VANT en vuelo al aire libre.

de ampliar la utilidad y demostrar algunos posibles usos que se pueden hacer con este tipo de vehículos [5-9].

Aprendimos acerca de la importancia del *software* libre y de la colaboración entre desarrolladores y su impacto en el desarrollo de nuevos proyectos, y que estén al alcance de todos para promover la investigación y la creatividad.

Gracias a esto fuimos capaces de crear un proyecto similar a los desarrollados por las grandes compañías, que en algún momento parecen inalcanzable, pero queda demostrado que siempre existen otras alternativas que dan resultados iguales o mejores.

En el ámbito de la programación nos instruimos en el funcionamiento de *software* de modelado y de presentación de modelos en 3D, cómo se cargan los modelos, texturas, cómo se mueven, se trasladan y se escalan.

También estudiamos y comprendimos a profundidad los datos contenidos en los archivos con extensión .OBJ y .MTL y cómo utilizarlos en diferentes proyectos.

En el área de realidad aumentada, aprendimos a utilizar la detección de patrones con la librería ARToolkit e incorporar los modelos digitales 3D a imágenes reales y cargarlos con OpenGL.

El VANT cuenta con un sistema de video incorporado, con el cual transmite video

en tiempo real a un computador, donde la imagen captada es procesada por un programa de realidad aumentada en búsqueda de un marcador, y a través de la librería de OpenGL, se colocan los modelos 3D sobre el marcador para su visualización.

Por otra parte, a lo largo de este tiempo se presentaron algunas dificultades como fallas técnicas y de configuración que se resolvieron poco a poco a través de la investigación y la experimentación.

Todo lo mencionado anteriormente permitió la culminación de este proyecto cumpliendo con todos los objetivos que se plantearon al inicio.

Como todo proyecto busca ser mejor con cada desarrollo, se recomienda a los futuros interesados en realizar un proyecto de este tipo, optimizar y mejorar el sistema de realidad aumentada por medio de mejoras a la cámara incorporada en el ámbito de resolución y enfoque para así poder reconocer patrones a una mayor distancia y reducir el ruido presente en la imagen.

También se podrían incluir más modelos 3D, otros tipos de marcadores para la detección de patrones o incluso realizar la proyección de modelos a través de detección de coordenadas GPS y orientación.

Otra recomendación sería mejorar algunos de los componentes físicos del vehículo aéreo no tripulado como añadirle una batería más grande, de manera que aumente su autonomía de vuelo, además es posible incorporar motores a la cámara para así brindar mayor movilidad y presentar un mayor número de ángulos de visión.

Como el VANT utiliza radiofrecuencia para transmitir el video y también se usa para su control, este sistema puede que presente problemas en un ambiente donde también se utilice radiofrecuencia para otros propósitos, por esto recomendamos que se busquen otros medios para transmitir como el *bluetooth*, *wifi*, y otros.

REFERENCIAS

[1] Barrientos A. "Vehículo aéreo no tripulado para uso civil. Tecnología y Aplicaciones." 2007.

- [2] Lizbeth Heras Lara, José Luis Villareal Benítez, "La realidad Aumentada: Una tecnología en Espera de Usuarios". Revista Digital Universitaria UNAM. 2007.
- [3] Mayank Suhirid, Kiran B Ladhane, Mahendra Singh, Vishwas A Sawant, "Lateral Load Capacity of Rock Socketed Piers Using Finite Difference Approach", Scientific & Academic Publishing, Journal of Civil Engineering Research, vol.1, no.1, pp.1-8, 2011.
- [4] Agostina Siracusa, "Realidad Aumentada, Recreando Patrimonio Histórico", Universidad de Palermo, Buenos Aires, Argentina, 2013.
- [5] Kevin R. Fall, W. Richard Stevens, TCP/IP Illustrated, Volume 1: The Protocols, 2nd ed., Addison-Wesley, USA, 2011.
- [6] Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification, IEEE Std. 802.11, 1997.
- [7] Mohamed Almorsy, John Grundy and Amani S. Ibrahim, "Collaboration-Based Cloud Computing Security Management Framework", in Proceedings of 2011 IEEE 4th International Conference on Cloud Computing, pp. 364-371, 2011.
- [8] A. Karnik, "Performance of TCP congestion control with rate feedback: TCP/ABR and rate adaptive TCP/IP", M. Eng. thesis, Indian Institute of Science, India, 1999.
- [9] J. Padhye, V. Firoiu, D. Towsley, "A stochastic model of TCP Reno congestion avoidance and control", Univ. of Massachusetts, Tech. Rep. 99-02, 1999.