Universidad Zaragoza
1542

Trabajo Fin de Grado

Inicialización del estado de un sistema visual-inercial.
State initialization for a visual-inertial system.

Autor/es

Jorge Vizárraga Huerta.

Director/es

Javier Civera Sancho.

Escuela de Ingeniería y Arquitectura
2017/2018

Inicialización del estado de un sistema visual-inercial.

Resumen.

En la tecnología SLAM, *"Simultaneous location and mapping"*, encontramos una de las tecnologías más útiles y punteras del momento. Lo innovador de esta tecnología reside en su capacidad de dar a una máquina el conocimiento, de manera instantánea, de su posición. Este conocimiento lo obtiene mediante la estimación de mapas 3D que le permiten conocer el entorno por el que se mueve con elevada precisión, sin la necesidad de dispositivos GPS, evitando los problemas de localización que aparecen dentro de edificios cerrados.

Uno de los sistemas utilizados, para el conocimiento de la posición y el mapeo del entorno, es el de los sistemas visuales-inerciales. Estos sistemas se basan en la combinación de señales visuales, captadas por una, o varias, cámaras con señales inerciales recogidas por un sensor inercial IMU.

En este TFG se va abordar uno de los principales problemas que afrontan este tipo de sistemas, la inicialización. Se vana a estudiar las soluciones al problema propuestas en [1] y en [2] y se van a implementar las ecuaciones planteadas en dichos trabajos.

El algoritmo creado busca que el sistema conozca, con el menor error posible, su posición a través de una recreación del entorno. A diferencia de la mayoría de sistemas actuales se requiere que se haga sin una semilla inicial previa, es decir, se busca una inicialización correcta del sistema sin la utilización de un estado inicial dado.

Para evaluar el algoritmo implementado se ha utilizado el dataset EuRoC [6]. Este dataset contiene por un lado las imágenes tomadas por dos cámaras y, por otro, las señales inerciales recogidas por un sensor IMU durante el vuelo de un dron a través de una habitación. Además, incluye unas medidas ground truth que permiten comparar los resultados obtenidos con los resultados reales, pudiendo así estimar el error presentado por el algoritmo en la determinación de la posición.

Con diferentes experimentos se ha querido comprobar si el sistema estudiado puede llevar a cabo una inicialización correcta, o muy cercana a la correcta. Para este análisis se ha obtenido el error obtenido con respecto al ground truth.

Este estudio ha querido también analizar la robustez y la eficiencia del sistema planteado en [1] y [2]. Se ha estudiado su robustez, es decir, como reacciona el sistema ante situaciones cambiantes y se han hecho análisis de sensibilidad de las variables que más influyen en el resultado final.

TRABAJOS DE FIN DE GRADO / FIN DE MÁSTER

Escuela de Ingeniería y Arquitectura Universidad Zaragoza

DECLARACIÓN DE AUTORÍA Y ORIGINALIDAD

(Este documento debe acompañar al Trabajo Fin de Grado (TFG)/Trabajo Fin de Máster (TFM) cuando sea depositado para su evaluación).

D./Dª. Jorge Vizárraga Huerta,

con nº de DNI 25358806H en aplicación de lo dispuesto en el art. 14 (Derechos de autor) del Acuerdo de 11 de septiembre de 2014, del Consejo de Gobierno, por el que se aprueba el Reglamento de los TFG y TFM de la Universidad de Zaragoza,

Declaro que el presente Trabajo de Fin de (Grado/Máster) Grado, (Título del Trabajo) Inicialización del estado de un sistema visual-inercial

es de mi autoría y es original, no habiéndose utilizado fuente sin ser citada debidamente.

Zaragoza, 17/09/2018

Jorge Vizarraga

Fdo: Jorge Vizárraga Huerta.

AGRADECIMIENTOS

En primer lugar quiero darle las gracias a mi tutor Javier Civera Sancho, como alumnos de último año de grado nos encontramos en una etapa crucial en nuestras vidas y que una persona con su conocimiento, prestigio y ocupación haya querido colaborar conmigo es algo que no olvidaré fácilmente. Gracias por haberme inculcado tantos valores positivos y gracias por poner tu granito de arena en mi formación.

Gracias por guiarme como lo has hecho y, sobretodo, gracias por tu eterna paciencia.

No puedo concluir esta sección sin mencionar a mis padres, por su apoyo incondicional y por haberme empujado y educado para ser la persona que soy hoy.

Nada hubiera sido posible sin estas personas.

GRACIAS

Índice.

1. Introducción.

La visión por ordenador es una disciplina científica cuyo objetivo es proporcionar a una máquina el sentido humano de la vista. Si se define al sentido de la vista como aquel que permite al ser humano crear en el cerebro una recreación de nuestro entorno se debe recordar que cualquier ser o sistema, que pretenda utilizar el mismo, deberá ser capaz de, autónomamente, llevar a cabo todas aquellas tareas que permiten al ojo y al cerebro humano llevar a cabo esta función.

Una de esas tareas es a la que se dedica la parte de la visión por ordenador que se va a abordar en este trabajo, esta parte es la referente a lo conocido como "structure from Motion". Esta técnica permite la estimación de estructuras 3D a partir de secuencias de imágenes 2D y señales inerciales medidas por un sensor IMU. Esta técnica permite replicar el funcionamiento del cerebro humano, en el cual se obtiene una recreación del entorno a través de la combinación de las imágenes proporcionadas por la vista con las medidas del movimiento relativo, movimiento entre el propio humano y el entorno que le rodea.

En este trabajo se aborda uno de los principales problemas de estos sistemas visuales-inerciales, que es la inicialización del estado. Tradicionalmente, al estudiar estos sistemas se parte de un estado inicial ya dado o supuesto.

Dado un vehículo aéreo de pequeño tamaño "MAV", al que se le aplica un sistema de este tipo, la necesidad de una semilla inicial a la hora de reiniciar el sistema impide totalmente el uso del mismo en vuelos de mediana y larga duración. Esto adquiere una importancia mayor cuando se tiene en cuenta que, en este tipo de vuelos, es frecuente que surjan fallos que requieran de una rápida y eficiente reiniciación en carencia de un estado inicial.

La importancia fundamental del estado inicial en este tipo de sistemas de fusión de señales visuales e inerciales reside en que éstos funcionan con algoritmos iterativos, es decir, dado un estado previo y medidas sensoriales son capaces de devolver un estado actualizado. Esto se traduce en que si el estado inicial contiene errores se van a corromper todos los estados posteriores dándole al MAV una recreación del entorno errónea que puede desembocar en una colisión, con el coste económico que ello conlleva.

Es por lo expuesto anteriormente que este TFG se centra en el estudio de la solución a dicho problema planteada en [1] y [2], artículos muy recientes y estado del arte del tema. En esta solución se propone un sistema en el que se fusionan datos visuales e inerciales para obtener la estructura del entorno a escala global sin la necesidad de una semilla o estado previo.

Esta tecnología tiene diversas aplicaciones como, por ejemplo, los coches de conducción autónoma, lo cuales necesitan ser totalmente conscientes de su entorno en todo momento y, obviamente, reiniciar su estado en ausencia de una semilla inicial si lo necesitan. Otra aplicación, qué es hacia la cual va orientada este TFG, es a los MAV con

navegación autónoma. Este sistema es de muy interesante aplicación a éstos debido a que es capaz de dar una respuesta rápida sin necesidad de un procesador muy potente y, como gran aliciente, es capaz de devolver la posición sin necesidad de una señal GPS lo cual permite a estos vehículos transitar por el interior de edificios.

El objetivo de esta investigación es determinar si, con las ecuaciones planteadas en [1] y [2], es posible crear un algoritmo capaz de devolver la posición del MAV sin necesidad de una semilla previa y con el error mínimo posible.

Para mejorar la eficiencia del sistema se van a llevar a cabo análisis de sensibilidad de varios parámetros con el objetivo de ver como se puede obtener el funcionamiento más óptimo del sistema.

Para lograr dichos objetivos se ha seguido el siguiente proceso:

1º Estudio de las ecuaciones de [1] y [2].

2º Obtención de los datos necesarios:

- Datos inerciales : Medidas de movimiento recogidas por un sensor IMU, proporcionado por el EuRoC dataset [6], a lo largo de una secuencia de imágenes.
- Datos visuales: Uso en *Matlab* del algoritmo de Lucas-Kanade.

3º Implementación del sistema matricial propuesto en [1] y [2] para obtener los resultados necesarios que permitan determinar la posición del MAV en diferentes secuencias de imágenes.

4º Optimización del tiempo de ejecución del sistema a través del análisis de ciertas variables del sistema.

5º Comparación del resultado obtenido con el del "*ground truth*" ofrecido en el dataset para así poder estudiar el error.

6º Análisis y toma de conclusiones de los resultados obtenidos.

2. Inicialización del estado visual-inercial.

2.1 Fundamentos teóricos.

Se considera un sistema visual-inercial compuesto de una cámara monocular y de un sensor IMU "*Inertial Measurement Unit*". Éste último es el encargado de proporcionar las medidas inerciales, que son la velocidad angular y aceleración lineal .

Se trabaja en un sistema de referencia global W, para lograrlo se aplicarán matrices de cambio de la referencia, como se indicará posteriormente en el apartado 2.3 de esta memoria.

El IMU proporciona medidas de velocidad angular y de aceleración lineal. La cámara extrae N puntos característicos en cada imagen, los cuales seguirá a lo largo de toda la secuencia. En la figura 1 se puede observar una representación del sistema mencionado.

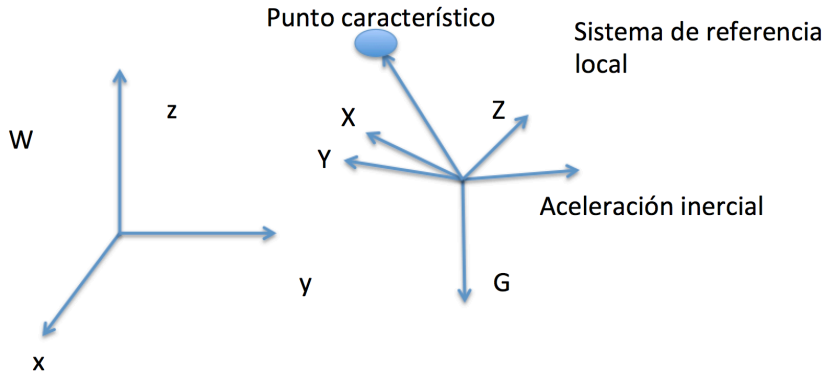

Figura 1. Sistemas de referencia.

El sistema, cámara-IMU, opera durante un intervalo corto de tiempo. Durante este tiempo la cámara observa "ni" imágenes y "N" puntos característicos , los cuales seguirá a lo largo de todas las imágenes. El algoritmo solo trabaja con aquellos puntos que la cámara es capaz de encontrar en todas las imágenes.

En [1] y en [2] se propone la siguiente ecuación:

$$S_j = \lambda_1^i \mu_1^i - V t_j - G \frac{t_j^2}{2} - \lambda_j^i \mu_j^i \qquad \{1\}$$

En la que, habiendo j = 1….ni fotos e i = 1….N puntos característicos.

- S_j es la integración de las medidas inerciales del IMU en la imagen j.
- μ_j^i es la orientación del punto i en la foto j.
- t_j es el tiempo en el que ocurre la foto j.
- V es la velocidad inicial del sistema.
- G es el vector gravedad en el instante inicial.
- λ_j^i es la distancia entre el sistema y el punto i en la foto j.

En [1] encontramos la figura 2, que es una representación gráfica de la ecuación {1}.

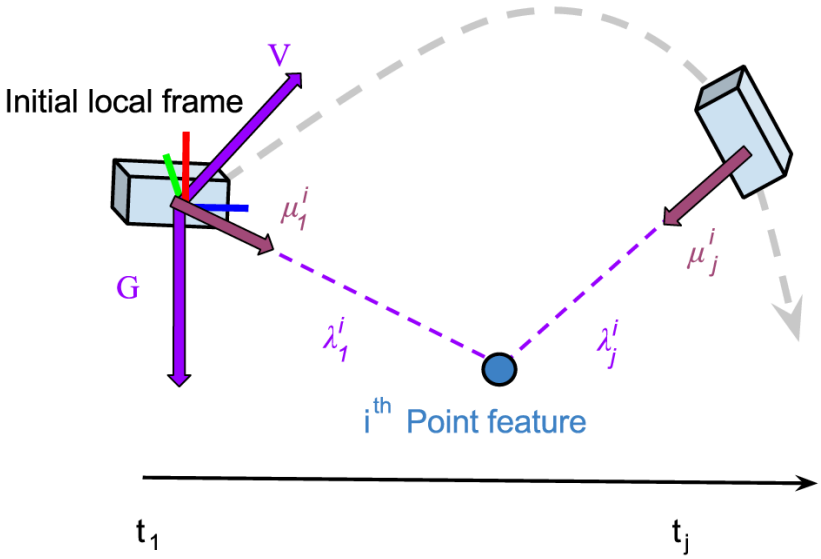

Figura 2. Representación gráfica de {1}. Figura extraida de [1].

La orientación de cada punto en cada instante, el tiempo y la integración de las medidas inerciales son todos conocidos ya que se obtienen como combinación de las medidas giroscópicas y visuales, en el caso de la orientación, y de las medidas inerciales y giroscópicas en el caso de S.

Siendo la distancia a cada punto, la gravedad y la velocidad inicial las incógnitas de {1} habrá, por un lado, 6 incógnitas, formadas por los 3 componentes del vector G y otros 3 del vector V, y por otro, las incógnitas correspondientes a la distancia de cada punto en cada imagen trabajada.

Se establece un sistema matricial con 3 ecuaciones por punto, esto es debido a que tanto la orientación de cada punto como su distancia al sistema tiene 3 componentes (x,y,z). Es necesario saber que en este sistema se empieza a calcular desde la segunda imagen, ya que la imagen (1) se toma como referencia.

Por lo tanto, el sistema propuesto tiene (6 + ni*N) incógnitas y (3*(ni-1)*N) ecuaciones. El sistema matricial se puede observar a continuación, siendo los términos que lo componen:

- S la matriz de la integración inercial
- X la matriz de incógnitas.
- $T_j = -\frac{t_j^2}{2} I_3$.
- $S_j = -t_j I_3$.
- $I_3 =$ La matriz identidad 3x3.
- $O_3 =$ La matriz nula 3x1

$$\Xi X = S. \quad \{2\}$$

$$S \equiv [S_2^T, \dots, S_2^T, S_3^T, \dots, S_3^T, \dots, S_{ni}^T, \dots, S_{ni}^T]^T$$

$$X \equiv [G^T, V^T, \lambda_1^1, \dots, \lambda_1^N, \dots, \lambda_{ni}^1, \dots, \lambda_{ni}^N]^T$$

$$\Xi = \begin{bmatrix} T_2 & S_2 & \mu_1^1 & O_3 & O_3 & -\mu_2^1 & O_3 & O_3 & O_3 & O_3 & O_3 \\ T_2 & S_2 & O_3 & \mu_1^2 & O_3 & O_3 & -\mu_2^2 & O_3 & O_3 & O_3 & O_3 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ T_2 & S_2 & O_3 & O_3 & \mu_1^N & O_3 & O_3 & -\mu_2^N & O_3 & O_3 & O_3 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ T_{ni} & S_{ni} & \mu_1^1 & O_3 & O_3 & O_3 & O_3 & O_3 & -\mu_{ni}^1 & O_3 & O_3 \\ T_{ni} & S_{ni} & O_3 & \mu_1^2 & O_3 & O_3 & O_3 & O_3 & O_3 & -\mu_{ni}^2 & O_3 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ T_{ni} & S_{ni} & O_3 & O_3 & \mu_1^N & O_3 & O_3 & O_3 & O_3 & O_3 & -\mu_{ni}^N \end{bmatrix}$$

Finalmente, conocidas todas las incógnitas del sistema matricial, podemos obtener la posición del sensor en cada instante j a partir de la ecuación {2}, con la expresión {3}.

$$P_j = \frac{1}{N} \Sigma_i (\lambda_1^i \mu_1^i - \lambda_j^i \mu_j^i) \qquad \{3\}$$

Conocida la posición del sistema en cada instante j se puede comprobar, a través del ground truth proporcionado por el EuRoC dataset [6], si la posición es la correcta o no.

Para resolver el sistema matricial mencionado en el apartado 3.1 se ha utilizado la herramienta informática Matlab. En esta sección de la memoria se va a explicar, con

detalle, de que forma se ha obtenido cada una de las partes del sistema necesarias para resolver el mismo.

2.2 Extracción y seguimiento de los puntos característicos.

Para obtener los puntos característicos en la secuencia de imágenes se ha utilizado una herramienta de Matlab llamada "*VisionPointTracker*".

Esta herramienta se implementa en un código en el que lo primero que se hace es inicializar la herramienta para la primera imagen. Esta acción hace que se obtenga la posición de los puntos característicos en la primera imagen de la secuencia, lo que marcará cuáles van a ser los puntos a emparejar en el resto de la misma. En la figura 3 se pueden observar los puntos detectados por el seguimiento en una primera imagen ejemplo.

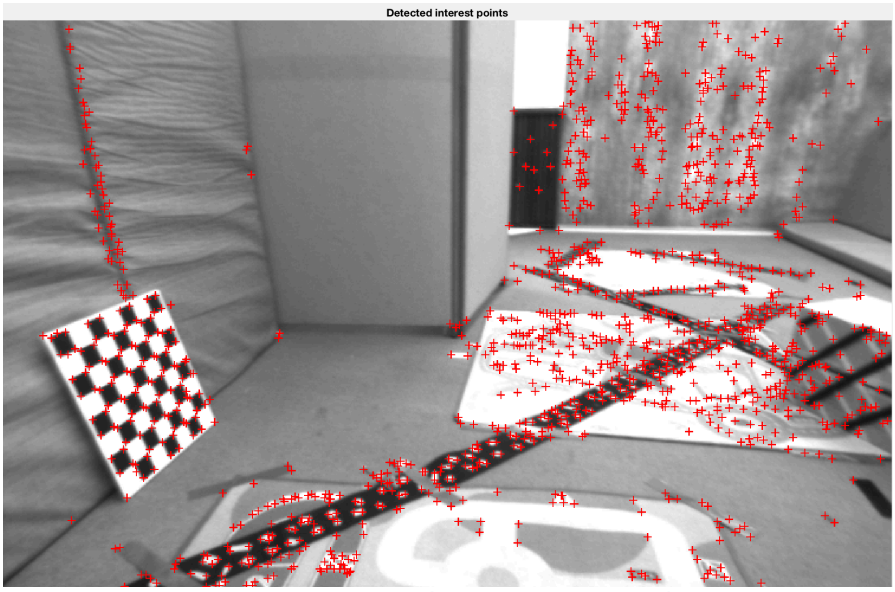

Figura 3. Extracción de puntos característicos.

Los detectores extraen los puntos característicos dentro de una imagen, es decir, aquellos pixel en el que hay un cambio de contraste como, por ejemplo una esquina. Para cada punto se calcula un descriptor, que se usará para seguir el punto en el resto de las imágenes.

Una vez que se han detectado los puntos característicos se aplica el seguimiento a toda la secuencia de imágenes. El seguimiento funciona de tal manera que, de una imagen previa a la siguiente, busca emparejar los puntos característicos obtenidos en la inicialización. Para que un punto sea emparejado con éxito debe cumplir el criterio del máximo error bidireccional. La manera en la que funciona este criterio es la siguiente:

1º Se encuentra un punto, presente en una imagen j-1, en la imagen actual j.

2º Se busca dicho punto de la imagen j en la imagen j-1.
3º Si el punto buscado coincide con el original del primer paso con una tolerancia menor al límite de pixeles establecido, entonces el punto se considera como válido.

En la figura 4 se observa una representación gráfica del proceso previamente descrito.

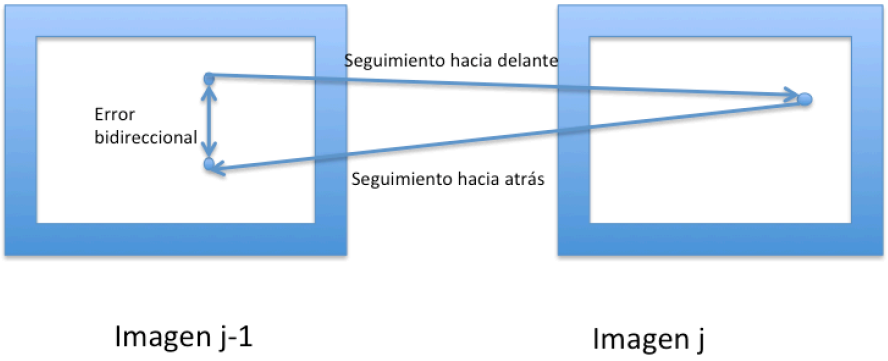

Figura 4. Criterio del error bidireccional.

Previamente a ejecutar el seguimiento con todos los puntos se comprueba que éste está realizando un seguimiento adecuado de cada punto. Para ello seguimos uno de los puntos emparejado exitosamente durante toda la secuencia y observamos su posición en 4 momentos de la secuencia distintos y muy alejados entre sí.

En la figura 5(a) se observa dicho punto en la primera imagen de la secuencia, en la 5(b) se observa el mismo punto en la imagen nº14, en la 5(c) en la 30 y, finalmente, en la 5(d) se observa el punto en la ultima imagen de la secuencia. Como se puede comprobar el punto ha sido seguido correctamente durante toda la secuencia de imágenes.

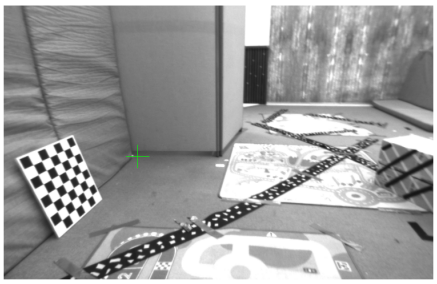

Figura 5(a) . Seguimiento en imagen 1.

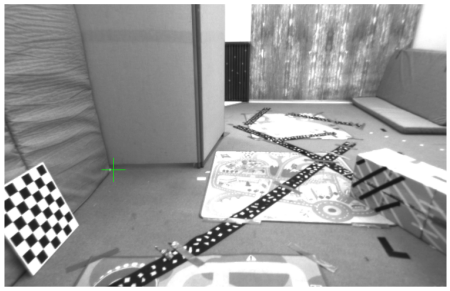

Figura 5(b). Seguimiento en imagen 14.

Figura 5(c). Seguimiento en imagen 30.

Figura 5(d). Seguimiento en la ultima imagen .

Una vez comprobado que los puntos son seguidos con éxito se procede a ejecutar el programa para todos los puntos válidos de la secuencia.

Al observar los resultados se observa que, de los N puntos característicos que el seguimiento había encontrado en su inicialización, sólo un número menor de puntos, N_pv < N, ha sido emparejado correctamente durante toda la secuencia. En el caso del ejemplo, en la inicialización encuentran 434 puntos de los cuales sólo 117 acaban siendo válidos. Más adelante en la memoria se demostrará como un numero así de puntos validos es más que suficiente para lograr un resultado preciso.

2.3 Obtención de la orientación de cada punto en cada instante.

Para obtener la orientación de cada punto en cada imagen se ha utilizado la siguiente ecuación:

$$\mu_j^i = R_j * R_{IMU \rightarrow CAM} * k^{-1} * \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \qquad \{4\}$$

Siendo:

R_j = La matriz de rotación en cada imagen j. Se obtiene a través de los ángulos "Roll, Pitch y Yaw" del MAV en cada instante. Estos ángulos son conocidos como los ángulos de navegación, o de Euler, y representan la orientación de un objeto, durante su navegación, en 3 dimensiones. Son muy útiles cuando se tiene un sistema de coordenadas móvil respecto de uno fijo y se desea dar la posición del sistema móvil en un momento dado, durante la trayectoria. En la figura 6 se pueden observar dichos ángulos, los cuales se obtienen como el producto de la velocidad angular en cada instante por el incremento de tiempo desde el instante previo. La velocidad angular viene dada por el IMU

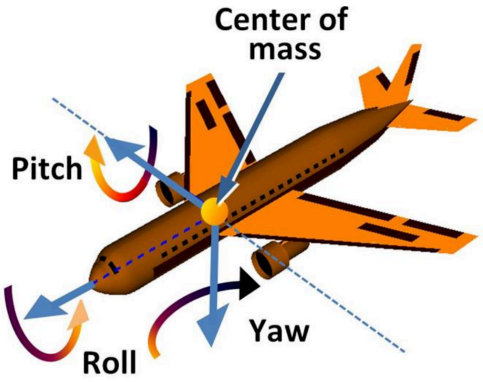

Figura 6. Ángulos de Euler. Figura extraída de [5].

Se le va a prestar especial atención al significado físico de la matriz de rotación, obtenida a partir de los ángulos roll, pitch y yaw, explicados previamente.

Como se ha explicado previamente, un cuerpo puede rotar sobre 3 ejes ortonormales, estas rotaciones son medidas por los 3 ángulos de Euler. Cada uno de estos 3 ángulos puede ser expresado como una matriz de rotación tal que:

- El yaw es la rotación $\propto$ alrededor del eje z, constituyendo una matriz de rotación 2D alrededor de los ejes "x" e "y", como se puede observar a continuación:

$$R_z(\propto) = \begin{pmatrix} cos \propto & -sen \propto & 0 \\ sen \propto & cos \propto & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

- El pitch es la rotación β alrededor del eje y, constituyendo una matriz de rotación 2D alrededor de los ejes "x" y "z", como se puede observar a continuación:

$$R_y(\beta) = \begin{pmatrix} cos\,\beta & 0 & sen\,\beta \\ 0 & 1 & 0 \\ -sen\,\beta & 0 & \cos\beta \end{pmatrix}$$

- El roll es la rotación γ alrededor del eje x, constituyendo una matriz de rotación 2D alrededor de los ejes "y" y "z", como se puede observar a continuación:

$$R_z(\gamma) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\gamma & -sen\,\gamma \\ 0 & sen\,\gamma & \cos\gamma \end{pmatrix}$$

Estas 3 matrices de rotación 2D pueden conformar una matriz de rotación 3D, llevando a cabo el producto de las mismas, esta matriz sirve para localizar un cuerpo 3D en cualquier orientación. La forma de esta matriz se puede observar a continuación.

$$R(\propto, \beta, \gamma) =$$

$$\begin{pmatrix} cos \propto cos\,\beta & cos \propto sen\,\beta\, sen\,\gamma - sen \propto \cos\gamma & cos \propto sen\,\beta\, cos\,\gamma + sen \propto \mathrm{sen}\,\gamma \\ sen \propto cos\beta & sen \propto sen\,\beta\, sen\,\gamma + cos \propto \cos\gamma & sen \propto sen\,\beta\, cos\,\gamma - cos \propto \mathrm{sen}\,\gamma \\ -sen\,\beta & sen\,\gamma\, cos\,\beta & cos\,\gamma\, cos\,\beta \end{pmatrix}$$

$R_{IMU \rightarrow CAM}$= Es la matriz de cambio de referencia necesaria para que la cámara trabaje en el mismo sistema de referencia que la IMU y, de esta manera, el sistema de referencia de la IMU y el de la cámara sean el mismo. Esto es necesario para que el sistema se encuentre trabajando en un único sistema de referencia. La matriz de cambio de referencia de la cámara a la IMU esta constituida por los extrínsecos de la cámara. Los parámetros extrínsecos corresponden a vectores de rotación y traslación que traducen las coordenadas de un punto 3D a un sistema de coordenadas.
Éstos conforman la matriz 3x3, tal que:

$$\begin{pmatrix} 0.0149 & -0.999 & 0.0041 \\ 0.9996 & 0.015 & 0.0257 \\ -0.0258 & 0.0038 & 0.9997 \end{pmatrix}$$

k^{-1} es la matriz inversa de la matriz de calibración de la cámara tal que:

$$k = \begin{pmatrix} F_u & 0 & C_u \\ 0 & F_v & C_v \\ 0 & 0 & 1 \end{pmatrix}.$$

Fu, Fv, Cu y Cv son los parámetros intrínsecos de la cámara, siendo Fu y Fv las componentes de la distancia focal de la cámara, en pixeles. Los parámetros Cu y Cv miden la intersección, en pixeles, del eje óptico con el plano de imagen desde el origen de la imagen.

En la figura 7 se puede apreciar un resumen de lo explicado previamente sobre la utilidad de los parámetros, extrínsecos e intrínsecos, de la cámara.

Figura 7. Parámetros extrínsecos e intrínsecos.

$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix}$ = Hace referencia a las coordenadas del punto i en la imagen j, obtenidas con la herramienta de seguimiento.

2.4 Obtención del vector S.

El vector S corresponde a la preintegración de las medidas inerciales tomadas por el IMU, las cuales son las medidas correspondientes al acelerómetro y al giróscopo. Como se ha indicado antes, con las velocidades angulares se obtienen las matrices de rotación para cada instante. Con las aceleraciones lineales y las matrices de rotación para cada instante se puede obtener la velocidad del sistema en cada instante tal como se indica en la ecuación {5}.

$$V_j = V_{j-1} + R_{j-1} * a_{j-1} * \Delta t \quad \{5\}$$

Conocida la velocidad del sistema se puede obtener la preintegración de las medidas inerciales, utilizando la ecuación cinemática de la posición para movimiento con aceleración, una vez conocida la velocidad en cada imagen {6}.

$$S_j = S_{j-1} + R_{j-1} * V_{j-1} * \Delta t + 0{,}5 * R_{j-1} * a_{j-1} * (\Delta t)^2 \quad \{6\}$$

2.5 Eliminación del sesgo.

En [1] se explica como el sesgo, o error de medida, presente en el giróscopo del IMU afecta, de forma considerable, a la actuación del método propuesto en [2]. En el nuevo método presentado en [1] se estima el sesgo del giróscopo automáticamente y es, por tanto, un método robusto a dicho error, presentando resultados como los obtenidos en ausencia de BIAS y evitando así la necesidad de inversión en un giróscopo de alta calidad.

El dataset EuRoC [6] nos da el BIAS presente en las medidas del giróscopo y el presente en las medidas del acelerómetro. Para tenerlo en cuenta, antes de operar con las velocidades angulares y las aceleraciones lineales, restamos el sesgo a dichas magnitudes, de esta forma se está teniendo en cuenta el error en la medida.

En [1] se demuestra como el error del giróscopo es mucho mas relevante que el error del acelerómetro, de todas maneras, se elimina el sesgo en ambos para dar el mejor resultado posible.

2.6 Construcción y optimización del sistema matricial.

Al ser un sistema diseñado para funcionar en cualquier momento de la trayectoria, la inicialización ha de ser lo más rápida posible. Para ello se ha prestado especial atención a la forma en la que se ha construido la matriz Ξ. Como se ha indicado antes, ésta, es una matriz muy dispersa ya que la gran mayoría de sus términos son 0. Debido a esto la forma en la que se ha afrontado la construcción de esta matriz ha sido una forma directa y no iterativa, forma que obligaría al programa a recorrer una gran cantidad de celdas de valor nulo.

Como se ha indicado previamente, las dimensiones de la matriz Ξ son (6 + ni*Npv) columnas y (3*(ni-1)*Npv) filas. Debido a la elevada dimensión de ni y Npv, el tamaño del sistema matricial se puede volver tan excesivo que, o bien requiera de un procesador de elevada capacidad o, de no tenerlo, provoque que el sistema no trabaje de una manera lo suficientemente rápida para ser considerado eficiente. La solución que se ha planteado para resolver este problema consiste en:

- Trabajar sólo con los puntos de mejor calidad, es decir, los que presenten un valor en su métrica superior a un limite establecido.
- Encontrar el número mínimo de imágenes para el cual el sistema sea capaz de dar el funcionamiento esperado del mismo.

Se han llevado a cabo dos experimentos, uno en el que se ha medido el tiempo de ejecución del programa en función del número de imágenes y otro en función del número de puntos.

En la figura 8 se expone la relación del numero de imágenes, eje x, con el tiempo de ejecución del programa, eje y . Se ha hecho para un número fijo de puntos, 117, y se

puede comprobar como al reducir el numero de imágenes se pueden llegar a reducir el tiempo hasta en un 24 %, tiempo que puede ser clave en este caso.

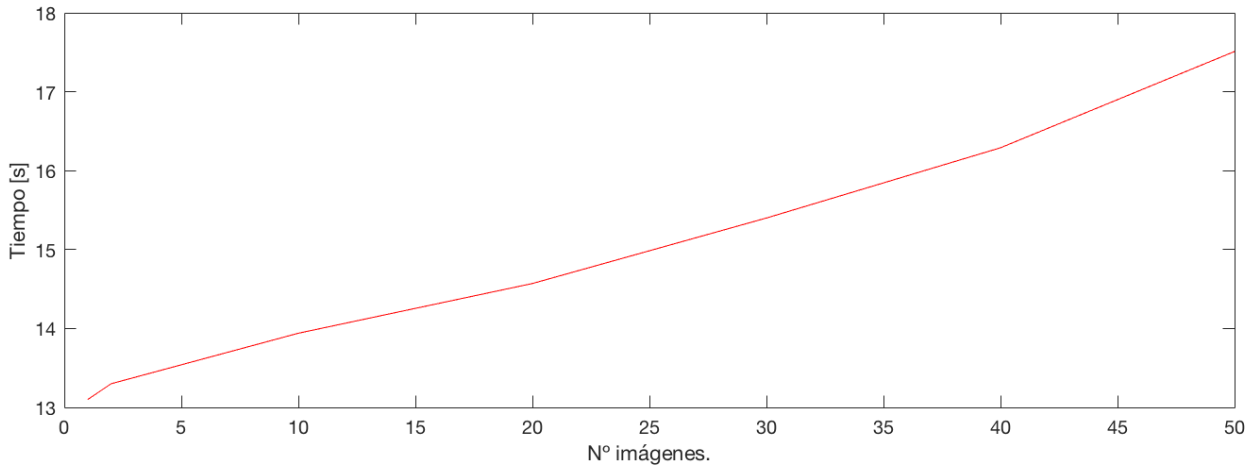

Figura 8. Relación entre el número de imágenes y el tiempo.

En la figura 9 se expone la relación del numero de puntos, eje x, con el tiempo de ejecución del programa, eje y. Se demuestra como, cuanto menor sea el número de puntos utilizados, el tiempo de ejecución del programa disminuye. Se ha experimentado con un número de 50 imágenes fijo.

Tras la realización de estos experimentos ha quedado expuesto que el programa dará su mejor rendimiento cuándo trabaje con el mínimo número de puntos e imágenes que garanticen un cálculo de la posición correcto.

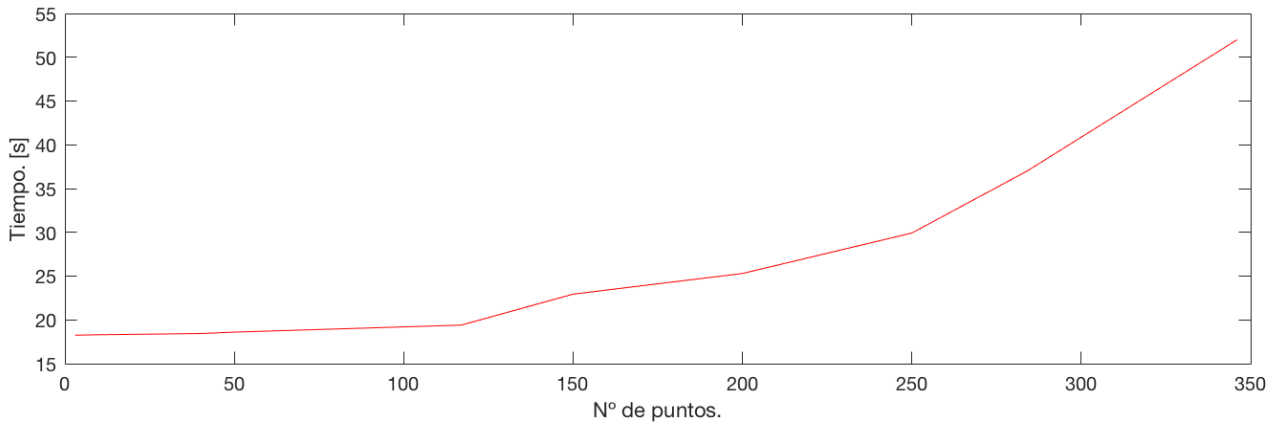

Figura 9. Relación entre el número de puntos y el tiempo.

3. Obtención del resultado y análisis del mismo.

Para la resolución del sistema explicado en el apartado 2 se ha utilizado un ordenador Macbook Pro con 8 GB de memoria RAM y un procesador Intel Core i5. Se ha ejecutado el sistema sobre la base de datos EuRoC [6] de la cual se ha obtenido la posición del sistema para diferentes inicializaciones y se han analizado los comportamientos de diferentes variables con el objetivo de optimizar el funcionamiento del sistema.

Con el cálculo del error en la trayectoria se han podido sacar diferentes conclusiones sobre como mejorar el funcionamiento del sistema y con el tiempo de ejecución se ha podido llevar a cabo, a través de la ejecución de diversos experimentos, una labor de mejora en lo que concierne a la eficiencia del sistema.

De forma adicional, a lo hablado en los papers estudiados, se han llevado a cabo experimentos para determinar la influencia que tiene el número de imágenes, empleados en las secuencias, sobre la capacidad del sistema para calcular el modulo de la gravedad de forma correcta.

3.1 Base de datos EuRoC.

La base de datos EuRoC [6] utiliza un dron Asetec FIrefly hex-rotor, que se puede observar en la figura 10. Para futuras referencias, la base de datos está referenciada en [6].

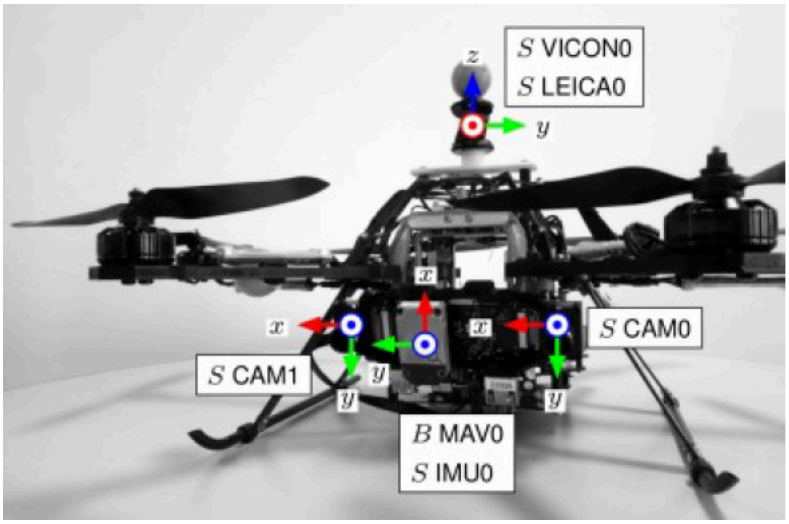

Figura 10. MAV utilizado en EuRoC. Figura extraída de [6].

Este sistema consiste en dos cámaras monocromáticas (2x20 FPS, 20 Hz) que, como se acaba de indicar, grabar secuencias con una frecuencia de 20 Hz, un sensor IMU, que contiene un acelerómetro y un giróscopo, que le permiten tomar medidas de velocidad angular y aceleración lineal. El sensor IMU toma medidas con una frecuencia de 200 Hz y, por último, también cuenta con un sistema de captura del movimiento con

6 grados de libertad, esto permite obtener el movimiento real de la cámara y la estructura real de la escena, ground truth. El ground truth se utilizará, en este apartado, para comparar los resultados obtenidos por el sistema con los resultados reales. Esto brinda la oportunidad de calcular el error en el cálculo y, con este conocimiento, poder analizar el sistema estudiado con el objetivo de maximizar el rendimiento del mismo.

Como se ha mencionado, el IMU tiene una frecuencia de 200 Hz y toma medidas sincronizadas a las de la cámara, lo cual permite fusionar datos visuales con inerciales.

Para los experimentos se ha utilizado el sensor IMU0 y la cámara CAM0. Como se puede apreciar en la figura 19 la cámara utilizada y el IMU se encuentran en dos sistemas de referencia distintos, es por ello que, tal y como se ha explicado en el apartado 2, a la hora de realizar los experimentos se ha utilizado la matriz de cambio de referencia R_{IMUCAM} para que IMU y cámara trabajen en el mismo sistema de referencia.

Para construir dicha matriz se ha utilizado el dataset, del cual se han sacado los extrínsecos cámara-IMU, los intrínsecos de la cámara y la alineación espacio-temporal real.

Para sincronizar las señales es necesario entender que el IMU toma una señal cada 5 ms y la cámara toma una foto cada 50 ms, lo que significa que a la hora de hacer la preintegración de una imagen a otra se utilizan 10 señales de IMU. Para lograr una sincronización satisfactoria ha sido necesario identificar en el dataset los timestamps coincidentes para cada una de las secuencias analizadas.

Es necesario recordar que, como se ha explicado en el apartado 2.5, se han eliminado de las medidas del IMU los sesgos presentes en el giróscopo y en el acelerómetro. Los sesgos se obtienen del dataset, sincronizados a 200 Hz con las medidas del IMU.

Se han utilizado las imágenes de la habitación Vicon 1, esta habitación contiene diversos objetos repartidos por la habitación para determinar la eficacia del sistema a la hora de obtener los puntos característicos. Se pueden analizar estas imágenes en 3 niveles de dificultad distintos, fácil, medio y difícil. Se ha optado por la utilización del nivel fácil, nivel que carece de demasiados movimientos bruscos y cambios de luz, situaciones que representan limitaciones en el funcionamiento de este sistema, como se indicará posteriormente. En la figura 11 se puede apreciar una de las múltiples vistas de la habitación.

Figura 11. Habitación Vicon 1. Figura extraída de [6].

3.2 Eliminación de la distorsión.

Antes de ejecutar el sistema y de analizar los resultados se han modificado las imágenes, proporcionadas por las cámaras del sistema, utilizando la función de Matlab "undistortimage" de tal manera que se ha eliminado la distorsión presente en las imágenes.

Este paso en el proceso es de elevada relevancia debido a que si la imagen se encuentra distorsionada el seguimiento va a acumular un error en la localización de los puntos característicos que corromperá el resultado final. En la figura 12 se puede observar a la izquierda la imagen distorsionada y a la derecha la imagen una vez corregida.

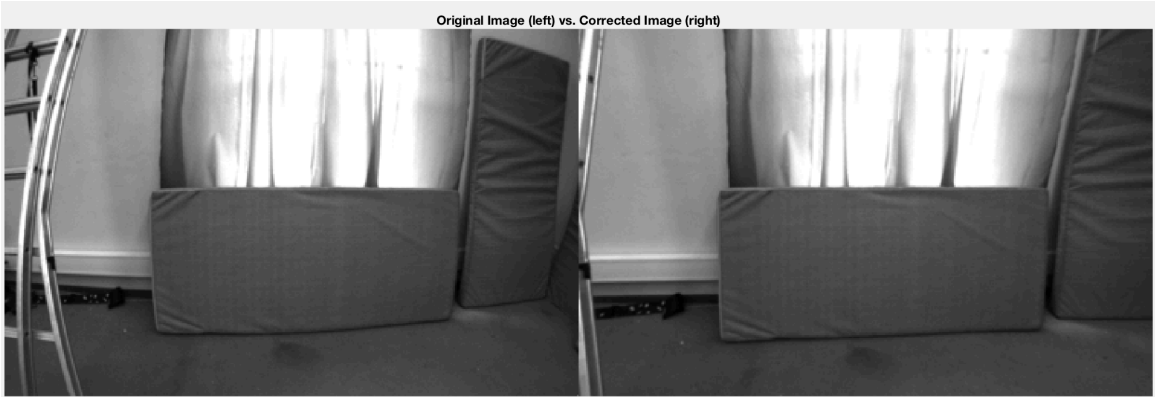

Figura 12. Correción de la distorsión en una imagen ejemplo.

3.3 Error en la posición y limitaciones.

Al resolver el sistema se obtiene el vector X. Dicho vector contiene al vector gravedad, el vector velocidad y las profundidades de los puntos. Conocidas las profundidades de cada punto en cada imagen, y sus correspondientes orientaciones, se procede a utilizar la ecuación {3} para llevar a cabo el objetivo final de este trabajo, el calculo la posición del sistema en cada imagen. Una vez calculada ésta se compara con la posición, que nos da el ground truth del dataset EuRoC [6], y se comprueba el error en el calculo de la misma con la expresión {7}.

$$error(\%) = 100 * \left| \frac{\|P_{GT}\| - \|P_j\|}{\|P_{GT}\|} \right| \qquad \{7\}$$

Se han llevado a cabo un gran número de experimentos con diferentes secuencias del dataset y se han recogido y expresado los resultados del error en el histograma que se puede apreciar en la figura 13. En el eje x se observa el error(%) y en el eje y se observa la cantidad de veces que ha salido ese error en los experimentos.

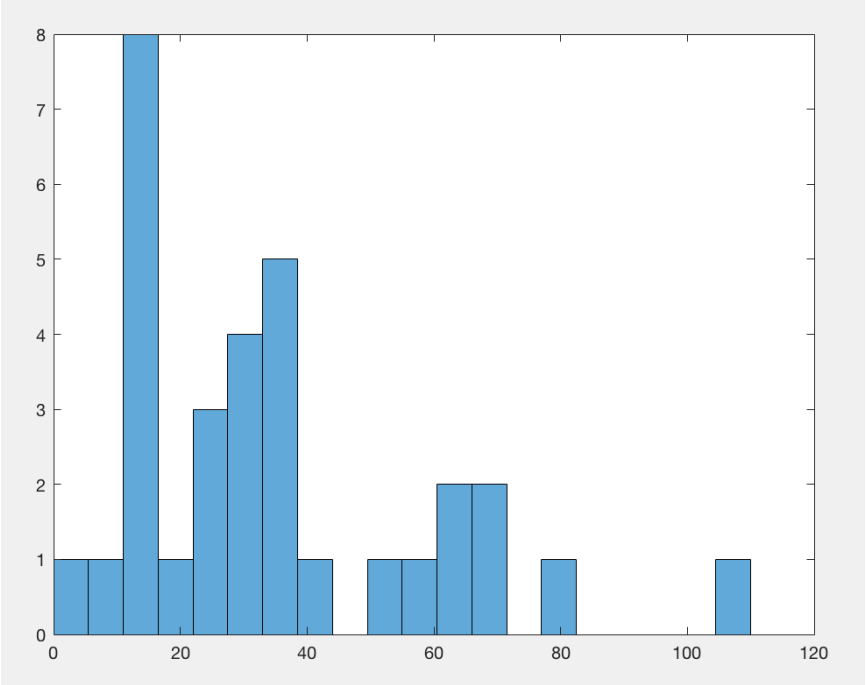

Figura 13. Histograma del error.

Como se puede observar en el histograma, en la mayoría de experimentos, se han obtenido errores en la estimación de la posición cercanos al 20%, lo cual es una estimación lo suficientemente precisa como para considerarla correcta, teniendo en cuenta que una estimación de la posición se suele considerar correcta para errores menores al 30%. El motivo de considerarlas correctas es que los algoritmos iterativos, que calculan posición mediante optimización no lineal, suelen converger para dichos valores de error.

Sin embargo, es necesario comprender las limitaciones del teorema que se está estudiando, las cuales provocan que el error haya resultado muy alto en ciertos experimentos. Para entender mejor las limitaciones se han dedicado los apartados siguientes a explicarlas y, también, a exponer ejemplos de experimentos en los que el resultado final se ve corrompido por las mismas. Al ser un sistema que funciona por la fusión de datos visuales con datos inerciales se va a abordar el tema desde las limitaciones de cada tipo de dato por separado para favorecer el entendimiento del problema.

3.3.1 Limitación inercial.

El sistema lleva a cabo la preintegración de los datos inerciales para la obtención del vector S, necesario para resolver el sistema final.

Para que la preintegración sea correcta es necesario que exista una cantidad de movimiento suficiente a lo largo de la secuencia para que el IMU pueda recoger datos fiables. En algunos experimentos se han usado secuencias en las que el sistema apenas se movía. Sí apenas hay variación de movimiento el sistema no es capaz de calcular la escala del entorno de manera correcta y, por lo tanto, no devuelve una posición correcta.

Además, si la secuencia es demasiado corta, el sistema no va a disponer de suficientes medidas inerciales con las que hacer una preintegración correcta. Para comprobar esto se han llevado a cabo experimentos en secuencias cuyos datos visuales ya se sabe que son buenos. El experimento consiste en ir bajando el tiempo de secuencia hasta que el sistema deje de estimar la posición de forma correcta.

Un buen ejemplo de secuencias con poco movimiento es la secuencia inicializada para la imagen 340, esta secuencia, a partir de 30 imágenes logra errores del 17%, lo que, en casos como este donde las distancias de traslación son bastante cortas, es considerado un resultado aceptable. Para menos de 30 imágenes el error oscila entre el 84% y el 40 %, demostrando que, con tan poco tiempo, la estimación es poco fiable. En las figuras 14 (a) y 14 (b) podemos observar el poco movimiento que se ha producido durante las primeras 25 imágenes. Nótese que puede haber secuencias en las que haya suficiente movimiento en las primeras imágenes como para llevar a cabo una preintegración correcta.

Figura 14(a). Imagen 1

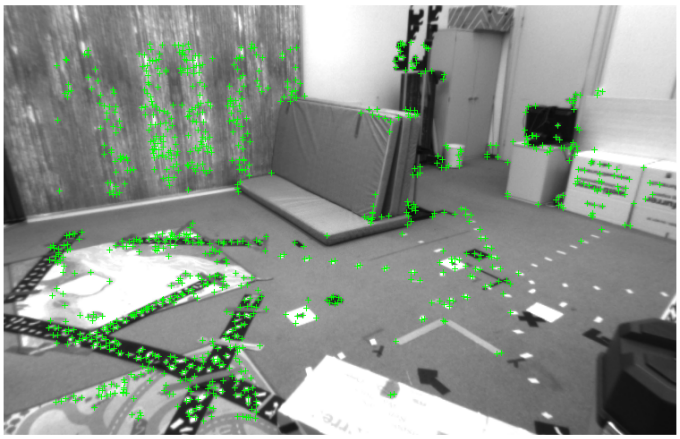

Figura 14(b). Imagen 25

Como solución a este problema se propone nunca trabajar con secuencias con menos de 15 segundos de duración, es decir, secuencias con menos de 30 imágenes. Aunque es cierto que hay secuencias que son capaces de dar resultados fiables para menos de 30 imágenes, lo que aquí se busca, debido a la importancia del correcto funcionamiento del sistema, es minimizar la posibilidad de fallo y, dado que, como se verá más adelante, esta no es la única potencial fuente de error, utilizar como mínimo secuencias de 15 segundos se considera una solución ventajosa al problema.

3.3.2 Limitación visual.

El sistema planteado se ve fuertemente limitado por la mayor limitación del seguimiento de puntos característicos. Esta limitación aparece cuando el sistema realiza un giro muy brusco y una parte, lo suficientemente grande, de la primer imagen deja de ser visible durante la secuencia. Ante esto el sistema devuelve un gran error en el calculo de la posición debido a su incapacidad de emparejar una gran parte de los puntos iniciales. Sí la escena cambia bruscamente entre la primera y la ultima imagen el sistema se va a encontrar con la fuerte limitación de que no va a buscar aquellos puntos característicos que definen a la nueva escena. Esto es debido a que el point seguimiento no ha sido inicializado para la nueva escena provocando que, al fusionar estos datos visuales erróneos con los inerciales, obtenidos de manera correcta, el sistema no funciona.

Para demostrar esto se ha utilizado el programa en la secuencia que cubre las fotos 200 a 240 en el dataset EuRoC [6]. En la figura 15 (a) se puede observar la primera imagen, para la cual ha sido inicializado el sistema, con un total de 280 puntos característicos marcados. En la figura 15 (b) se observa la última imagen de la secuencia, se puede apreciar un fuerte cambio de orientación que ha provocado que sólo se hayan podido emparejar puntos en la parte izquierda de la habitación, resultando en que, durante gran parte de la secuencia, una importante parte de la escena no está siendo estudiada visualmente, haciendo imposible una correcta estimación. Como era de esperar, en esta secuencia se obtiene un 80% de error en la estimación de la posición.

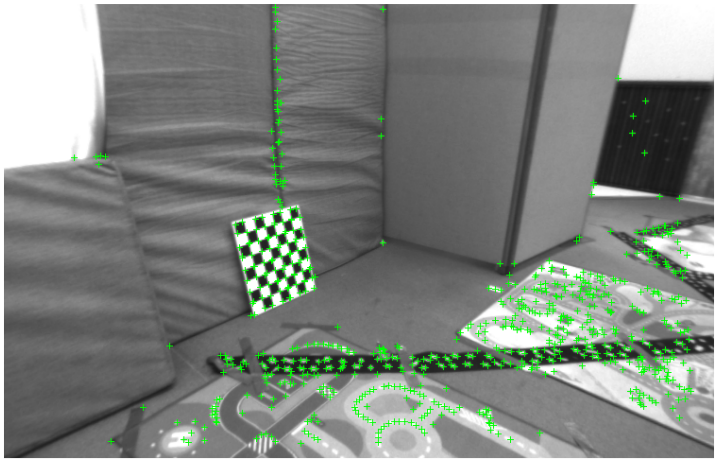

Figura 15(a) . Imagen de inicialización.

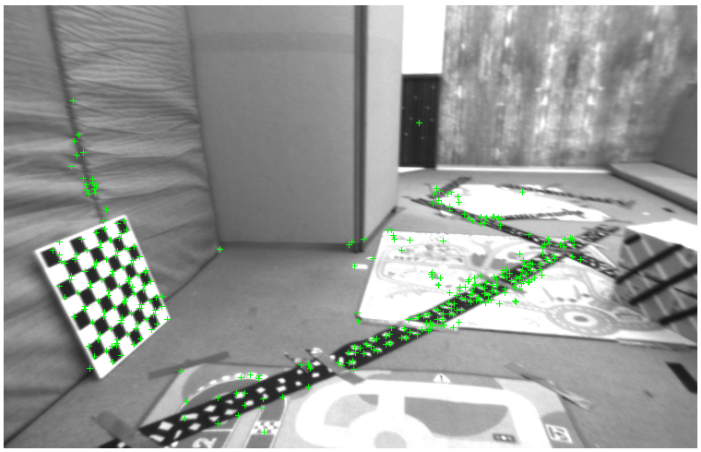

Figura 15(b). Última imagen de la secuencia.

Esta limitación del seguimiento también provoca que no se pueda trabajar con secuencias de imágenes excesivamente largas pues, en el momento en el que la imagen es lo suficientemente distinta a la de la inicialización, el sistema no funciona.

Para solucionar este problema se propone no trabajar con más de 30 imágenes. Pese a que existen casos en los que no se puede evitar que el sistema caiga en esta limitación, con 30 imágenes se cumplen las condiciones comentadas en 3.3.1, para evitar las limitaciones inerciales, y se suelen evitar las limitaciones visuales ya que para la mayoría de secuencias estudiadas las limitaciones visuales no aparecen con este número de imágenes.

3.3.3 Limitación en el número y la calidad de los puntos.

Una de las carencias que más afectan al sistema es la incapacidad, por parte del seguimiento utilizado, para descartar puntos de baja calidad. El seguimiento busca puntos característicos en cada imagen y guarda su posición y su saliencia. La saliencia es un dato que nos permite medir la calidad con la que se ha localizado un punto. Como se ha visto antes, un número excesivo de puntos resulta en un tiempo de ejecución no eficiente, es por esto por lo que siempre se va a intentar operar con el número mínimo de puntos que garantice una resultado satisfactorio.

Se han llevado a cabo experimentos en diferentes secuencias. Estos experimentos han consistido en aplicar un filtro, en la métrica, a los puntos emparejados inicialmente por el seguimiento. En estos experimentos se ha aprovechado el conocimiento de la métrica para exponer como se puede reducir el número de puntos hasta un valor límite sin desestabilizar demasiado el error en la estimación de la posición. Que el error se mantenga constante hasta un valor límite significa que, la eliminación con el filtro de aquellos puntos de métricas mas bajas, no afecta al resultado

pero si que reduce considerablemente el tiempo de ejecución del sistema. Tras varios experimentos se ha determinado un filtro medio de 0,008 con el que se han logrado resultados satisfactorios, en aquellos experimentos que cumplen con las condiciones necesarias para no ser afectados por las limitaciones expuestas anteriormente.

Se ha determinado también otra conclusión muy importante, en lo referente a la calidad de los puntos, al llevar a cabo estos experimentos. En esta investigación se ha visto como secuencias que, a priori, cumplían con las limitaciones inerciales y visuales no lograban devolver una estimación correcta de la posición. Al estudiar la calidad de los puntos se ha visto que, en estos casos, el seguimiento no es capaz de emparejar una cantidad suficiente de puntos de calidad. La calidad de un punto hace referencia al error bidireccional con el que ha sido encontrado un punto durante la trayectoria. Sí no se trabaja con suficientes puntos buenos los errores de los malos se irán acumulando, imposibilitando un cálculo correcto de la posición.

Esto resulta en que el sistema no sea capaz de calcular la posición, una vez aplicado el filtro, en secuencias, con una cantidad de movimiento suficiente y, en la inicialización, una gran cantidad de puntos, al no tener puntos buenos. Sí no se aplicara el filtro se obtendría un cálculo de la posición con errores no aceptables. En la figura 22 se puede ver como, tras aplicar el filtro previamente mencionado, desaparecen casi todos los puntos a lo largo de la secuencia. En la figura 16(a) se puede ver la última imagen de la secuencia sin filtro, en la 16(b) se puede ver como alrededor de 220 puntos no han superado el filtro. En esta secuencia, antes de aplicar el filtro, se estaba obteniendo un error cercano al 90%, lo que indica que la calidad de los puntos es muy restrictiva a la hora de evaluar el funcionamiento correcto del sistema.

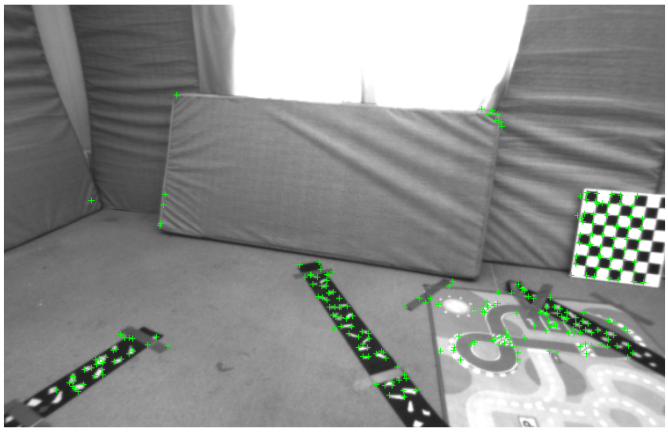

Figura 16(a). Última imagen en condiciones normales.

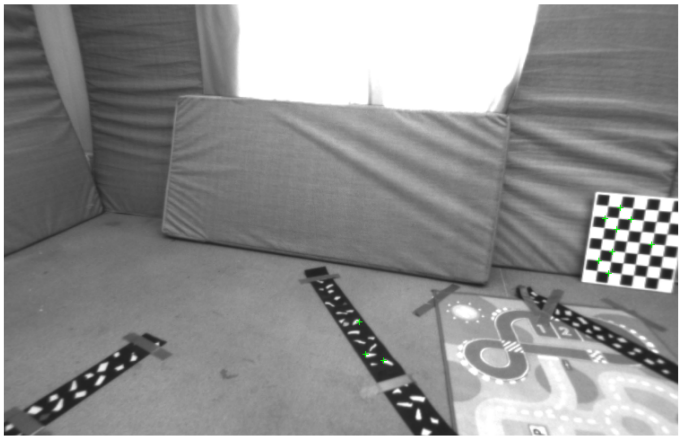

Figura 16(b). Última imagen tras aplicar el filtro

En la figura 17 se puede ver una representación de cómo el error se mantiene estable mientras se eliminan los puntos de baja calidad y de cómo al eliminar puntos consistentes el error va aumentando poco a poco hasta que no se tienen estimaciones aceptables de la posición.

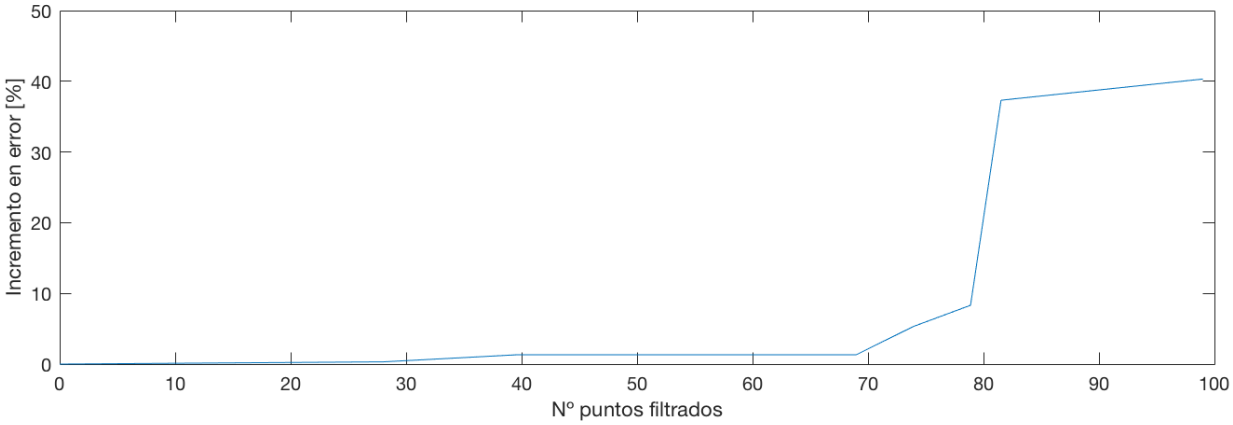

Figura 17. Incremento en el error en función del número de puntos filtrados.

En el eje x se observa el % de los puntos que han sido eliminados por el filtro y en el eje y se representa el incremento en el error, del calculo de la posición, con respecto al error obtenido sin aplicar el filtro.

Se observa como el error se mantiene estable, respetando una tolerancia del $\pm 1{,}5\%$, hasta que se han filtrado el 70 % de los puntos. Esto supone una optimización considerable en cuanto al tiempo de ejecución del sistema ya que indica que el programa va a dar el funcionamiento esperado con apenas el 35% de los puntos originales.

3.4 Correlación entre ni y N en el tiempo.

Como se ha explicado, en el apartado 3.3.2, un número excesivo de imágenes va a provocar que el sistema no funcione de forma correcta, siendo el límite máximo de imágenes aquel que garantice una secuencia en la que no se produzcan movimientos bruscos, movimientos que provoquen que el seguimiento no pueda hacer un emparejamiento efectivo de los puntos. Por debajo de ese número limite de imágenes ahora el interés se focaliza en evaluar cual es el número óptimo de imágenes con el que ejecutar el sistema. Hay que tener en cuenta que, cuantas mas imágenes se usen, menos puntos se emparejaran a lo largo de la secuencia y viceversa.

En el apartado 2.6 se ha explicado que, para cantidades elevadas de imágenes y de puntos, el tiempo de ejecución del programa es mayor. Es por esto por lo que se ha llevado a cabo un experimento en el que se ha evaluado, para diferentes secuencias, el tiempo de ejecución del programa, utilizando 30, 40 y 50 imágenes.

Se quiere comprobar si tiene más peso en el tiempo de ejecución un mayor número de puntos o un mayor numero de imágenes. Se ha hecho el experimento para decenas de secuencias distintas y se han representado los resultados en la figura 18. La curva roja hace referencia a los experimentos con 30 imágenes, la azul a aquellos con 40 imágenes y la verde a aquellos con 50 imágenes. El eje y representa el tiempo de ejecución y el eje x el número de puntos utilizado.

A partir de la gráfica se comprueba que, para un menor numero de imágenes, hay un mayor numero de puntos emparejados. Las pruebas con 30 imágenes suelen emparejar entre 350 y 700 puntos, las de 40 imágenes entre 530 y 100, finalmente, las de 50 suelen emparejar entre 30 y 300. Nótese que, independientemente del número de imágenes, siempre va a haber excepciones, con experimentos que apenas emparejan puntos, debido a las limitaciones inerciales y visuales.

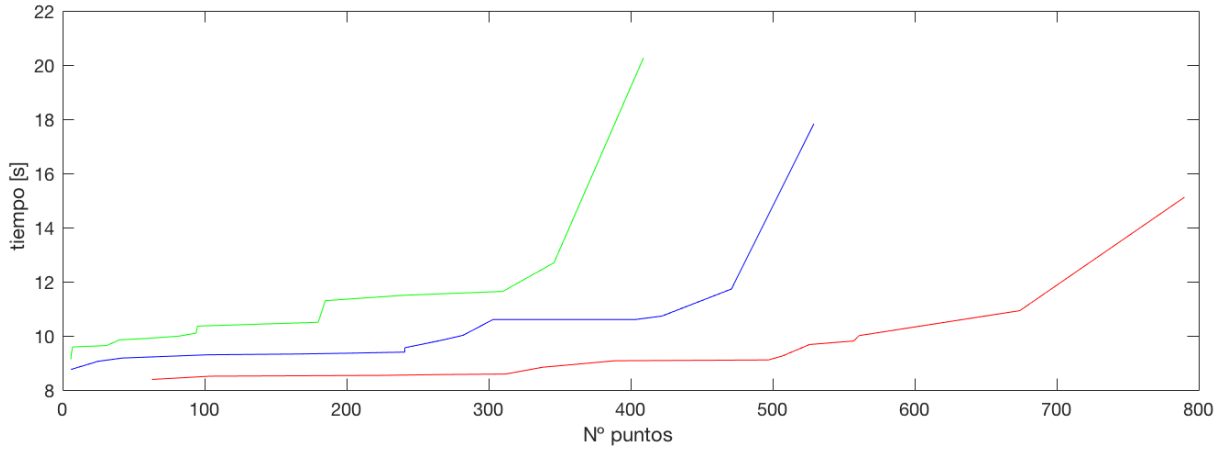

Figura 18. Tiempo de ejecución en función del Nº de puntos y el Nº de imágenes.

Habiendo hecho un estudio del número óptimo de imágenes, se puede concluir que, el número, ha de ser siempre menor que aquel que reduzca bruscamente el número de emparejamientos y siempre ha de haber un número suficiente de imágenes que garantice que el tiempo de secuencia sea el suficiente para que la preintegración inercial sea satisfactoria, es importante recordar que este sistema no funciona sólo con datos visuales.

Dentro de estos límites, sabemos que, a mayor número de imágenes, menor número de puntos. Siendo el tiempo de ejecución cuadráticamente proporcional a ambas variables hay que encontrar un compromiso, entre ambas variables, que asegure la calidad de la solución, dentro de un tiempo razonable.

3.5 Influencia del número de imágenes en el cálculo de la gravedad.

La ecuación planteada {1} refleja como el sistema matricial calcula las profundidades de cada punto en cada imagen, necesarias para calcular la posición del sistema, pero también calcula la gravedad y la velocidad del sistema en el instante inicial.

Aunque no tan importantes como las profundidades, fundamentales a la hora de sacar la posición, la velocidad y la gravedad también son de importante análisis pues son un fuerte indicativo de lo correcto del funcionamiento del sistema. Si se estudia detenidamente la ecuación V y G y se extraen sus términos, éstos se calculan tal que:

$$S_j = -Vtj$$
$$S_j = -G\frac{t_j^2}{2}$$

De esas relaciones se puede concluir que V y G se calculan directamente desde la preintegración de los datos realizada para calcular S_j. Al ser la gravedad en la tierra $9{,}8\,\frac{m}{s^2}$ se puede comparar el modulo de la gravedad obtenida con la gravedad en la tierra para saber si el sistema está funcionando de forma correcta y para evaluar otras cosas como el número mínimo de imágenes o de puntos a partir del cual estima la gravedad correctamente, lo cual es condición necesaria pero no vinculante para el calculo correcto de la posición.

Para estudiar la influencia del número de imágenes en la gravedad se van a llevar a cabo simulaciones del sistema, se han llevado a cabo diversas simulaciones del sistema, simulaciones que se ha comprobado que respetan las limitaciones del sistema, se ha hecho con este tipo de secuencias para que el error acumulado por las limitaciones no contamine los valores medios del módulo de gravedad que se obtienen con un funcionamiento correcto y para, de esta forma, poder relacionar cómo afecta el número de imágenes en el cálculo de la gravedad.

En la figura 19 se puede apreciar la gráfica que relaciona el número de imágenes, en el eje x, con el valor de la gravedad obtenido por el sistema. Se puede observar como a medida que va aumentando el número de imágenes, en secuencias

donde no se producen movimientos bruscos, ni existe falta de movimiento y donde la calidad de los puntos característicos es suficientemente buena, el valor de la gravedad se va estabilizando alrededor de 9,81, valor real de la gravedad.

Sin embargo, cuando se trabaja con números inferiores a 10 imágenes existe una especie de régimen transitorio en el cual el valor de la gravedad, aunque cercano al teórico, es muy volátil. Esto significa que el calculo de la gravedad es muy robusto ya que, en las condiciones de trabajo, le basta con poco mas de 5 imágenes para empezar a acercarse a valores cercanos al 9,81 buscado.

Analizando la ecuación {1}, la razón por la que con números tan reducidos de imágenes el sistema no es capaz de llevar a cabo una estimación correcta de la gravedad tiene una relación directa con el hecho de que, con números menores a 5 imágenes, el tiempo durante el que se lleva a cabo la preintegración es inferior a 250 ms. Este tiempo no constituye un tiempo suficiente para que el IMU sea capaz de tomar un número de medidas inerciales que habilite al sistema a dar un resultado positivo.

Cabe recordar que se está estudiando un sistema que combina medidas inerciales con medidas visuales, es por esta razón por la que resulta crucial que el sistema disponga del tiempo necesario para recoger suficientes medidas inerciales. Sí no hay suficientes medidas no se dispone de información suficiente sobre el movimiento del sistema.

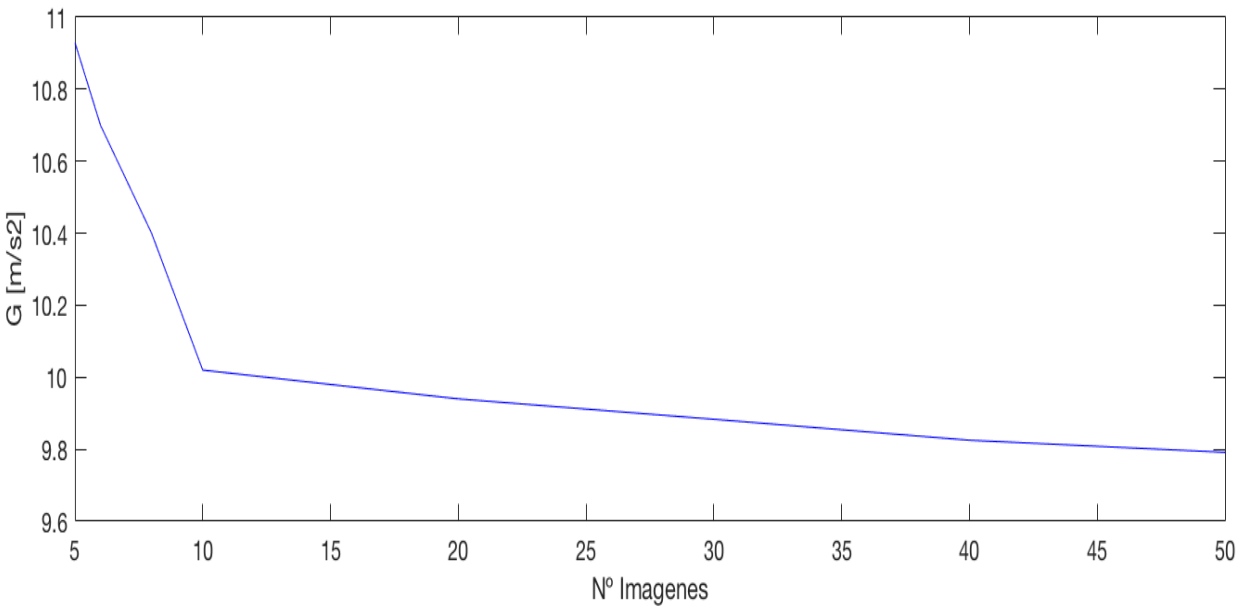

Figura 19. Módulo del vector Gravedad en función del Nº de imágenes.

4. Conclusiones

Este trabajo ha consistido en estudiar, implementar y analizar los algoritmos de los trabajos [1] y [2]. Ambos abordan el mismo problema, un problema muy recurrente en “structure from motion”, que es la inicialización de un sistema visual-inercial. Estos trabajos proponen unas ecuaciones, que proporcionan solución a la inicialización de un sistema sin necesidad de una semilla previa.

Una vez entendido el fundamento teórico del problema, aprendiendo el significado físico de sus ecuaciones y de los elementos que las forman, se ha desarrollado desde cero una implementación del sistema utilizando como herramienta de software el programa Matlab. Al ser un sistema donde intervienen muchas variables para dar un único resultado, la seguridad en el cálculo de cada variable ha sido crucial y, dicha labor, ha sido clave en la última etapa del trabajo para poder analizar como afecta cada variable al sistema.

El siguiente paso a la implementación del software ha sido ejecutarlo y evaluarlo en diferentes condiciones de trabajo, siempre con el objetivo de analizar su robustez a situaciones cambiantes y su eficacia. Para lograr esto se han llevado a cabo diferentes experimentos. Para poder llevara cabo experimentos con el programa se ha utilizado la base de datos EuRoC [6]. Dicha base de datos ha proporcionado los valores inerciales, velocidad angular y aceleración lineal, y visuales necesarios para probar el sistema.

Además, gracias a que dicha base de datos contenía un fichero ground truth, se ha podido comprobar la precisión en la determinación de la posición que ofrece el sistema, permitiendo obtener conclusiones sobre el funcionamiento final del sistema, que no es otro que determinar su posición.

Al existir ciertas situaciones en las que no se obtenía el resultado esperado, se han analizado las limitaciones que este sistema exhibe pudiendo determinar las causas por las que dichas limitaciones aparecen, esto permite añadir realismo a dos papers que son fundamentalmente teóricos.

Se ha estudiado también de que formas se puede mejorar la eficiencia del sistema, prestando especial atención a como reducir el tiempo de ejecución del sistema, el cual es una variable crucial en su aplicación comercial, en este caso el tiempo es dinero.

Los experimentos que se han llevado a cabo han sido:

- Obtención de la posición para diferentes secuencias y determinación de su error con respecto al del ground truth.
- Análisis de las limitaciones presentadas por el seguimiento , a la hora de recoger los datos visuales.
- Análisis de las limitaciones presentadas por el IMU, a la hora de recoger los datos inerciales.

- Determinación de cómo afecta el número de puntos utilizados al tiempo de ejecución del sistema.
- Determinación de cómo afecta el número de imágenes utilizados al tiempo de ejecución del sistema.
- Correlación entre el número de puntos y el de imágenes en su impacto sobre el tiempo de ejecución del sistema. Ver cual de las dos variables tiene más peso.
- Análisis del impacto de la presencia de puntos de baja calidad en el sistema.
- Influencia del número de imágenes en la estimación de la gravedad. Esta parte nos es mencionada en los papers [1] y [2] pero su análisis ha sido considerada de fuerte relevancia.

Tras realizar todos estos experimentos se ha llegado a las conclusiones que, pese a haber sido mencionadas durante esta memoria, van a ser comentadas en este apartado.

El algoritmo visual-inercial creado permite la inicialización del sistema sin la necesidad de especificación de un estado, o semilla, previo. Esto significa que, al estar basado en medidas visuales e inerciales, se puede crear una nube de puntos en tres dimensiones de la escena sin tener conocimiento previo de la misma.

Con el primer experimento, en el que se compara la posición obtenida con la de ground truth, se ha concluido que el sistema no es perfecto, que existen ciertas limitaciones que empañan su uso, pero también que, en la mayoría de ocasiones, ha sido capaz de devolver una estimación de la posición con menos de un 30% de error, lo cual se considera una estimación correcta.

Al analizar las limitaciones del seguimiento de Lucas Kanade utilizado se ha concluido que su principal limitación aparece cuando la secuencia sufre un movimiento brusco en el que la imagen cambia mucho con respecto a la imagen con la que se había inicializado. Esto resulta en que la gran mayoría de puntos para los que se había inicializado no pueden ser emparejados y el resultado se ve perjudicado. Hay que evitar movimientos bruscos o secuencias con demasiadas imágenes.

Se ha determinado experimentalmente que grandes cantidades de puntos e imágenes perjudican al tiempo de ejecución del sistema, perjudicando su eficacia y reduciendo sus aplicaciones comerciales. Esto se debe a que, al manejar un sistema matricial tan grande, el número de puntos e imágenes es proporcional al tamaño de la matriz.

En lo que refiere al tiempo de ejecución del sistema se ha determinado que el número de puntos desciende con el número de imágenes y viceversa, se ha visto como el número de imágenes afecta mas al tiempo total de ejecución que el número de puntos por lo que el objetivo principal, como se ha comentado antes, es trabajar con el

menor número de imágenes posible, esta es la solución fundamental al problema del tiempo de ejecución.

Se ha concluido que la eliminación de los puntos de baja calidad, a través de la aplicación de un filtro en su métrica, no afecta al error en la estimación de la posición pero si que reduce enormemente el tiempo de ejecución del sistema, añadiendo una solución más al problema del tiempo.

En cuanto a las limitaciones presentadas por el IMU, se ha concluido que se requiere de un número mínimo de imágenes en la secuencia que garantice un tiempo mínimo de recogida de señales, suficiente para obtener la escala global de la escena. También es importante tener en cuenta que no puede trabajar con un número excesivo de imágenes debido al problema de la acumulación de ruido en los sensores.

Se ha investigado sobre la influencia del número de imágenes necesario para llevar a cabo una estimación correcta de la gravedad, algo que no se había planteado en [1] y [2], y se ha determinado que para un número de imágenes superior a 10 el sistema es capaz de devolver una estimación correcta de la gravedad.

Con los resultados obtenidos en todos estos experimentos se puede concluir que se ha logrado implementar un algoritmo capaz de llevar a cabo una inicialización correcta bajo unas condiciones determinadas, lo que significa que es un programa que no es perfecto. Esto, en parte, se debe a que éste es un estudio muy reciente del que casi no es posible encontrar bibliografía y que aún se encuentra en desarrollo.

Bibliografía.

[1] Jacques Kaiser, Agostino Martinelli, Flavio Fontana, and Davide Scaramuzza. Simultaneous state initialization and gyroscope bias calibration in visual inertial aided navigation. IEEE Robotics and Automation Letters, 2(1):18–25, 2017.

[2] Agostino Martinelli. Closed-form solution of visual-inertial structure from motion. International Journal of Computer Vision, Springer Verlag, 2013.

[3] Calibración de la cámara https://www.aprenderpython.net/calibracion-la-camara-opencv/.

[4] Steven M. Lavalle, 2006, Cambridge University Press. Yaw, pitch and roll rotations.

[5] NASA, 21 January 2010, 23:28 (UTC), http://www.grc.nasa.gov/WWW/K-12/airplane/rotations.html

[6] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. Achtelik and R. Siegwart, *The EuRoC micro aerial vehicle datasets*, **International Journal of Robotic Research**, DOI: 10.1177/0278364915620033, early 2016.