



UNIVERSIDAD DE JAÉN
Escuela Politécnica Superior de Jaén

Trabajo Fin de Grado

GESTIÓN DE INFORMACIÓN PROCEDENTE DE DRONES

Alumno: Daniel de la Rosa de la Rosa

Tutor: Prof. D. Francisco R. Feito Higuera
Dpto: Departamento de Informática

Tutor: D. Juan Manuel Jurado Rodríguez
Dpto: Departamento de Informática



Universidad de Jaén
Escuela Politécnica Superior de Jaén
Departamento de Informática

Don Francisco Ramón Feito Higuera y don Juan Manuel Jurado Rodríguez, tutores del Proyecto Fin de Carrera titulado: Gestión de Información procedente de Drones, que presenta Daniel de la Rosa de la Rosa, autorizan su presentación para defensa y evaluación en la Escuela Politécnica Superior de Jaén.

Jaén, Septiembre de 2018

El alumno:

Los tutores:

Daniel de la Rosa de la Rosa

Francisco Ramón Feito Higuera

Juan Manuel Jurado Rodríguez

ÍNDICE

Agradecimientos.....	7
ÍNDICE DE IMÁGENES	9
ÍNDICE DE TABLAS	13
RESUMEN	15
ABSTRACT	15
1. INTRODUCCIÓN	19
1.1. Motivación	21
1.2. Estado del arte	22
1.2.1. Primeros registros de uso de aeronaves no tripuladas y evolución durante el siglo XX.....	22
1.2.2. Primera y Segunda Guerra Mundial	23
1.2.3. Guerra Fría y finales de siglo	24
1.2.4. Uso de drones para estudios en los campos de la fotogrametría y geomática	25
1.3. Objetivos	28
1.4. Planificación	30
1.4.1. Identificación de tareas	30
1.4.2. Planificación temporal	31
1.5. Coste del proyecto	32
1.5.1. Costes hardware	32
1.5.2. Costes software	33
1.5.3. Costes humanos	33
1.5.4. Coste total.....	34
1.6. Estructura de la memoria	34
2. CONTEXTO ACTUAL EN EL USO DE DRONES.....	39
2.1. Regulación actual.....	40
2.2. Ventajas que provee la utilización de drones	41
2.3. Áreas de aplicación	42
2.4. Clasificación general atendiendo a su arquitectura	43
3. ADQUISICIÓN DE IMÁGENES	49
3.1. Introducción y primeras consideraciones.....	49
3.2. Dron y equipamiento utilizado	51
3.2.1. Dron	51
3.2.2. Sensor equipado	52
3.2.3. Aplicación para la planificación del vuelo	54

3.3.	Metodología seguida y configuración utilizada	56
4.	CLASIFICACIÓN Y SELECCIÓN DE IMÁGENES.....	61
4.1.	Problemas encontrados	61
4.2.	Exchangeable image file format (Exif)	62
4.3.	Lenguaje de programación y librería a utilizar	63
4.4.	Selección por altura.....	64
4.4.1.	Resultados	67
4.5.	Clasificación de imágenes por vuelo	69
4.5.1.	Resultados	72
5.	RECONSTRUCCIÓN DEL TERRENO	79
5.1.	Técnicas ‘Structure from Motion’ (SfM).....	79
5.1.1.	Visión estereoscópica y cámaras estéreo	79
5.1.2.	Algoritmos SfM.....	82
5.2.	Agisoft PhotoScan.....	84
5.2.1.	Python API	86
5.3.	Metodología seguida y resultados	91
5.3.1.	Primera prueba	91
5.3.2.	Primera ejecución con el conjunto de imágenes del terreno en obras (calidad media) 93	
5.3.3.	Segunda ejecución con el conjunto de imágenes del terreno en obras (calidad alta) 99	
5.3.4.	Ejecución con el segundo conjunto de imágenes (calidad muy alta)	101
6.	ÍNDICES DE VEGETACIÓN	109
6.1.	RVI.....	110
6.1.1.	Resultados	111
6.2.	NDVI	111
6.2.1.	Resultados	112
6.3.	GNDVI.....	113
6.3.1.	Resultados	113
7.	PROTOTIPO DE APLICACIÓN PARA VISUALIZAR LOS RESULTADOS.....	117
7.1.	Justificación	117
7.2.	Análisis de requisitos.....	118
7.2.1.	Requisitos funcionales	118
7.2.2.	Requisitos no funcionales	119
7.3.	Unity.....	119
7.4.	Aspectos del desarrollo	120

7.4.1.	Importación de modelo 3D	120
7.4.2.	Visualización de posiciones desde las que se tomaron las fotografías.....	124
7.4.3.	Otros aspectos.....	125
8.	CONCLUSIONES.....	129
8.1.	Conclusiones.....	129
8.2.	Trabajos futuros	129
9.	ANEXO	133
	BIBLIOGRAFÍA	135

Agradecimientos

Me gustaría expresar mis agradecimientos a todos los que han contribuido a que pudiera completar de forma satisfactoria estos fantásticos años de estudio y, finalmente, este trabajo del que me siento orgulloso:

Mi familia y mi novia, que siempre me ha apoyado, aconsejado y ayudado durante todos estos años. Todo ello además de alegrarse en mis mejores momentos y soportarme en los peores. Han sido mi pilar principal.

Todos mis amigos y compañeros que he tenido la oportunidad de conocer en esta etapa y con los que he compartido muchas horas de trabajo y tiempo de ocio. Especial mención a mis compañeros del 103, que desde el primer día me han enseñado lo que es el compañerismo y el trabajo en equipo en un ambiente “laboral”.

Como no, a mis tutores Francisco Feito y Juanma, que me han ayudado desde el comienzo del proyecto y han ejercido una gran paciencia conmigo, ayudándome a adentrarme en el mundo de los drones y en todos los aspectos posibles para la realización de este trabajo.

ÍNDICE DE IMÁGENES

Ilustración 1.1 – Estadísticas internacionales del sector de drones (2006-2016). Fuente: [1]	19
Ilustración 1.2 - Evolución del número de modelos de drones según su ámbito de aplicaciones a nivel mundial. Fuente: [1]	20
Ilustración 1.3 –Globos militares incendiarios austríacos cayendo sobre Venecia. Grabado en madera. Fuente: [5]	22
Ilustración 1.4 - Hewitt-Sperry Automatic Airplane, 1918. Fuente: [8]	24
Ilustración 1.5 - Dron OQ-2. Fuente: [9]	24
Ilustración 1.6 - Muestra de una de las fases de los algoritmos SfM. Se descompone una imagen dada (izquierda) en una serie de descriptores claves (derecha). Fuente: [13]	27
Ilustración 2.1 - Algunas clases de drones de rotor único. Fuente: [1]	43
Ilustración 2.2 - Algunas clases de drones multirrotor. Fuente: [1]	44
Ilustración 2.3 - Ilustración de dron híbrido VTOL de ala fija. Fuente: [1]	44
Ilustración 2.4 - Algunas clases de drones de ala fija. Fuente: [1]	45
Ilustración 3.1 - Image Capture Tips: Equipment and Shooting Scenarios. Fuente: [27]	50
Ilustración 3.2 - Mavic Pro. Fuente: [29]	52
Ilustración 3.3 - Parrot Sequoia. Fuente: [30]	53
Ilustración 3.4 - Pantalla de elección del tipo de misión en Pix4Dcapture. Fuente: Elaboración propia.	54
Ilustración 3.5 - Pantalla para configuración del área a abarcar con vista en modo satélite. Fuente: Elaboración propia.	55
Ilustración 3.6 - Pantalla para configuración del área a abarcar con vista en modo mapa 2D y detalle de los ajustes. Fuente: Elaboración propia.	55
Ilustración 3.7 - Pantalla de Pix4Dcapture en la que se pide seleccionar el dron utilizado de entre todos los compatibles. Fuente: Elaboración propia.	57
Ilustración 4.1 - Algunos de los datos Exif contenidos en una de las imágenes capturadas para este trabajo. Fuente: Elaboración propia.	63
Ilustración 4.2 - Muestra del dato Exif "GPS Altitude". Fuente: Elaboración propia.	65
Ilustración 4.3 - Pseudocódigo del script para el filtrado por altura a la que se tomaron las fotografías. Fuente: Elaboración propia.	66
Ilustración 4.4 - Resultados obtenidos aplicando el filtrado por altura al primer conjunto de imágenes, con una altura mínima indicada de 25 metros. Fuente: Elaboración propia.	68
Ilustración 4.5 - Muestra del dato Exif "Date/Time Original". Fuente: Elaboración propia.	69
Ilustración 4.6 - Pseudocódigo del script para la clasificación de las imágenes por vuelo. Fuente: Elaboración propia.	70
Ilustración 4.7 - Resultado de ejecución del script de clasificación aplicado sobre el primer conjunto de imágenes multiespectrales. Fuente: Elaboración propia.	72
Ilustración 4.8 - Estado de la carpeta antes de aplicar el script de clasificación, como se muestra en la Ilustración 4.7. Fuente: Elaboración propia.	73
Ilustración 4.9 - Estado de la carpeta después de aplicar el script de clasificación, como se muestra en la Ilustración 4.7. Fuente: Elaboración propia.	73
Ilustración 4.10 - Resultado de ejecución del script de clasificación aplicado sobre el primer conjunto de imágenes del dron. Fuente: Elaboración propia.	74
Ilustración 4.11 - Estado de la carpeta antes de aplicar el script de clasificación, como se muestra en la Ilustración 4.10. Fuente: Elaboración propia.	74

Ilustración 4.12 - Estado de la carpeta después de aplicar el script de clasificación, como se muestra en la Ilustración 4.10. Fuente: Elaboración propia.	74
Ilustración 5.1 - Ejemplo de configuración de cámaras estéreo. Fuente: [36]	80
Ilustración 5.2 - Ejemplo de imágenes estéreo. Como se puede observar, el árbol se encuentra mucho más cerca de las cámaras que la casa. Por ello, el desplazamiento del árbol de una fotografía a la otra, es mayor que el desplazamiento de la casa. Fuente: [37] .	81
Ilustración 5.3 - Ejemplo de imágenes estéreo (arriba) y mapa de disparidad resultante (abajo). Fuente: [35]	82
Ilustración 5.4 - Ejemplo del resultado de la correspondencia entre imágenes del algoritmo SfM. Fuente: [43].....	84
Ilustración 5.5 - Script que ejecuta el flujo básico de trabajo en PhotoScan. Fuente: Elaboración propia.	88
Ilustración 5.6 - Definición de la función "addPhotos" de la clase <i>Chunk</i> en la API de PhotoScan. Fuente: [45].....	89
Ilustración 5.7 - Resultado de prueba inicial. A la izquierda, nube de puntos dispersa; a la derecha, modelo 3D.	92
Ilustración 5.8 - Resultado en consola del cálculo de nube de puntos dispersa en el primer conjunto de imágenes. Fuente: Elaboración propia.	94
Ilustración 5.9 - Resultado del cálculo de nube de puntos dispersa en el primer conjunto de imágenes. Fuente: Elaboración propia.	94
Ilustración 5.10 - Resultado del cálculo de nube de puntos dispersa en el primer conjunto de imágenes, visto de frente. Fuente: Elaboración propia.	95
Ilustración 5.11 - Resultado en consola del cálculo de nube de puntos densa en el primer conjunto de imágenes. Fuente: Elaboración propia.	96
Ilustración 5.12 - Resultado del cálculo de nube de puntos densa en el primer conjunto de imágenes. Fuente: Elaboración propia.	96
Ilustración 5.13 - Resultado del cálculo de nube de puntos densa en el primer conjunto de imágenes, visto de frente. Fuente: Elaboración propia.	97
Ilustración 5.14 - Resultado en consola del cálculo del modelo 3D y textura en el primer conjunto de imágenes. Fuente: Elaboración propia.	97
Ilustración 5.15 - Resultado del cálculo de modelo y textura en el primer conjunto de imágenes. Fuente: Elaboración propia.	98
Ilustración 5.16 - Resultado del cálculo de modelo y textura en el primer conjunto de imágenes, visto de frente. Fuente: Elaboración propia.	98
Ilustración 5.17 - Resultados en forma de nube de puntos dispersa (izquierda), nube de puntos densa (centro) y modelo con textura (derecha) de la ejecución con parámetros de calidad alta sobre el conjunto de imágenes del terreno en obras, vistos desde arriba. Fuente: Elaboración propia.	99
Ilustración 5.18 - Resultados en forma de nube de puntos dispersa (arriba), nube de puntos densa (centro) y modelo con textura (abajo) de la ejecución con parámetros de calidad alta sobre el conjunto de imágenes del terreno en obras, vistos desde frente. Fuente: Elaboración propia.	100
Ilustración 5.19 - Resultado en forma de ortomosaico de la ejecución con parámetros de calidad alta sobre el conjunto de imágenes del terreno en obras. Fuente: Elaboración propia.	100
Ilustración 5.20 - Resultado en forma de MDE de la ejecución con parámetros de calidad alta sobre el conjunto de imágenes del terreno en obras. Fuente: Elaboración propia.	101

Ilustración 5.21- Resultado en consola del cálculo del modelo 3D y textura en el conjunto de imágenes de Torrecillas con calidad muy alta. Fuente: Elaboración propia.	102
Ilustración 5.22 - Resultados en forma de nube de puntos dispersa (izquierda), nube de puntos densa (centro) y modelo con textura (derecha) de la ejecución con parámetros de calidad muy alta sobre el conjunto de imágenes de Torrecillas, vistos desde arriba. Fuente: Elaboración propia.	102
Ilustración 5.23 - Resultados en forma de nube de puntos dispersa (arriba), nube de puntos densa (centro) y modelo con textura (abajo) de la ejecución con parámetros de calidad muy alta sobre el conjunto de imágenes del terreno de Torrecillas, vistos desde un lado. Fuente: Elaboración propia.	103
Ilustración 5.24 - Detalle del modelo y textura obtenidos como resultado de la ejecución con parámetros de calidad muy alta sobre el conjunto de imágenes del terreno de Torrecillas. Fuente: Elaboración propia.	104
Ilustración 5.25 - Fotografía de muestra del conjunto de imágenes del terreno de Torrecillas, que ilustra la dificultad del dron para captar la parte baja de los olivos. Fuente: Conjunto de imágenes utilizado.....	105
Ilustración 5.26 - Resultado en forma de MDE de la ejecución con parámetros de calidad muy alta sobre el conjunto de imágenes del terreno de Torrecillas. Fuente: Elaboración propia.	106
Ilustración 6.1 - Ventana que se muestra en PhotoScan al seleccionar en el menú "Herramientas" y, seguidamente, "Set Raster Transform". Permite realizar cálculos con las distintas bandas espectrales. Fuente: Elaboración propia.	110
Ilustración 6.2 - Resultado del cálculo RVI en el proyecto de imágenes multiespectrales. Fuente: Elaboración propia.	111
Ilustración 6.3 - Resultado del cálculo NDVI en el proyecto de imágenes multiespectrales. Fuente: Elaboración propia.	112
Ilustración 6.4 - Resultado del cálculo GNDVI en el proyecto de imágenes multiespectrales. Fuente: Elaboración propia.	113
Ilustración 7.1 - Ventana que muestra PhotoScan para la utilización de la herramienta de reducción de la malla. Fuente: Elaboración propia.	121
Ilustración 7.2 - Detalle de la malla antes de aplicar la reducción de polígonos.....	121
Ilustración 7.3 - Detalle de la malla después de aplicar la reducción de polígonos.	122
Ilustración 7.4 - Detalle distinto de la malla antes (izquierda) y después (derecha) de aplicar la reducción de polígonos. Fuente: Elaboración propia.	122
Ilustración 7.5 - Vista de malla después de aplicar la reducción de polígonos con el programa ZBrush. Fuente: Elaboración propia.	123
Ilustración 7.6 - Resultado de la ejecución del visualizado de posiciones virtuales desde las que se captaron las fotografías. Fuente: Elaboración propia.	125

ÍNDICE DE TABLAS

Tabla 1.1 - División de tareas del proyecto. Fuente: Elaboración propia.	32
Tabla 1.2 - Costes hardware. Fuente: Elaboración propia.	32
Tabla 1.3 - Costes software. Fuente: Elaboración propia.	33
Tabla 5.1 - Comparación de características entre las dos versiones disponibles de PhotoScan. http://www.agisoft.com/features/compare/	85

RESUMEN

El aumento del uso de los drones o vehículos aéreos no tripulados fuera de la industria bélica ha supuesto una revolución en una gran cantidad de estudios y otros sectores. Estos vehículos proveen versatilidad y, en general, muchas ventajas comparadas con otras herramientas tradicionales. Todo ello, además, reduciendo en muchos casos los costes. El campo de la fotogrametría aérea es una de estas áreas donde los drones han comenzado a establecerse. En ella, a partir de un conjunto de imágenes aéreas de un objeto o superficie, se es capaz de obtener un estudio de sus características y su modelo tridimensional. Los drones, por un lado, son capaces de facilitar en gran medida los procesos de captura de imágenes. Por otro lado, añaden aún más información y capacidad de estudio si se mezclan con el uso de sensores o microsensores.

En este trabajo, se afronta el reto de la gestión de información procedente de drones con el enfoque en este tipo de estudios multidisciplinarios. Específicamente, el estudio de superficies o terrenos. Para ello, inicialmente, se abordarán diversos conceptos que es necesario conocer, como la legislación vigente que regula el uso de drones o su papel específico en este área. Más tarde, se analizará la metodología seguida y procesos necesarios en el estudio. Todo ello con el objetivo de visualizar los desafíos presentes en el sector y proveer herramientas que puedan resolverlos de forma efectiva y eficiente, además añadir aun más utilidad.

ABSTRACT

The increase of use of drones or UAVs (Unmanned Aerial Vehicles) outside the war industry has been a revolution in a large number of studies and other sectors. UAVs provide versatility and, in general, many advantages compared to other traditional instruments. In addition, they may also provide costs reductions. The field of aerial photogrammetry is one of these areas where drones are being put in place. From a set of aerial images of an object or surface, it is possible to obtain, among other things, its 3D model. UAVs, on the one hand, are capable of facilitate the processes required on images capture. On the other hand, they add even more data and study capacity if they are combined with the use of sensors or microsensors.

In this work, the challenge of managing information from drones is addressed, focused on this type of multidisciplinary studies. Specifically, the study of terrains or surfaces. To achieve it, initially, it is needed to analyse some concepts like the current law on drones or their specific role in this area. Later, the methodology and necessary processes in aerial photogrammetry will be analysed. The ultimate goal is to visualize the challenges present in the sector and provide tools that can solve them effectively and efficiently, in addition to add even more utility.

Capítulo 1.

Introducción.

1. INTRODUCCIÓN

Los drones son vehículos aéreos no tripulados, de ahí que también se les llame VANT o UAV (Unmanned Aerial Vehicle) por sus siglas en inglés, que cada vez están siendo utilizados en más y variadas áreas. Desde sus comienzos en el campo militar, su uso en lugar de otras herramientas o técnicas tradicionales, en la mayoría de las ocasiones, proveen una reducción de costes importante, así como otras ventajas intrínsecas. Durante los últimos años, este sector ha experimentado una gran evolución en el mercado y, debido a la gran cantidad de aplicaciones posibles, junto a la enorme capacidad de expansión del sector, hacen que se prevea un notable crecimiento a medio plazo. Como muestra de ello, se adjuntan a continuación dos gráficos [1]: en la Ilustración 1.1 se aprecia el auge del número de fabricantes de drones, que se han llegado a triplicar en los últimos diez años, aumentando también cada vez más el número de países en los que se fabrican. Asimismo, en la Ilustración 1.2 se observa cómo, tras un inicio marcado por las aplicaciones en el sector militar, la evolución de los últimos años muestra que el gran aumento del número de modelos de drones ha recaído sobre las actividades civiles y mixtas, quedando reducidos a 600 modelos aquellos destinados a uso militar.

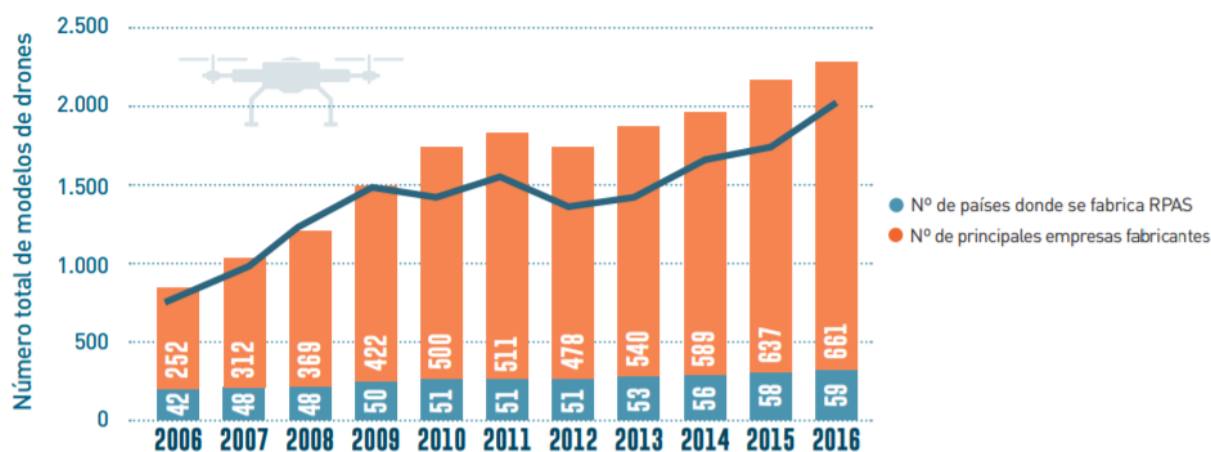


Ilustración 1.1 – Estadísticas internacionales del sector de drones (2006-2016). Fuente: [1]

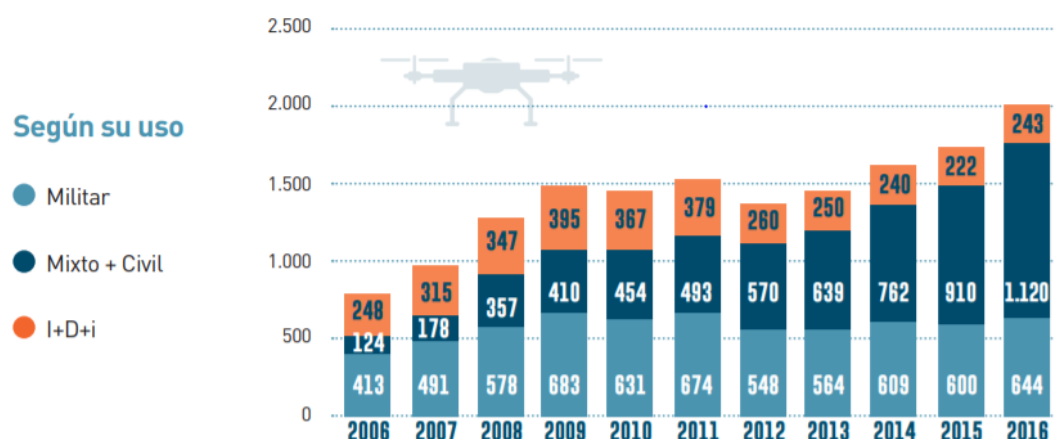


Ilustración 1.2 - Evolución del número de modelos de drones según su ámbito de aplicaciones a nivel mundial. Fuente: [1]

Una de las aplicaciones en la que los drones tienen una mayor proyección actual y futura es en el campo de la fotogrametría. Esta es la técnica cuyo objetivo es estudiar y definir con precisión la forma, dimensiones y posición en el espacio de un objeto cualquiera, utilizando esencialmente medidas hechas sobre una o varias fotografías de ese objeto. Específicamente, la fotogrametría aérea es aquella que usa imágenes tomadas desde el aire [2]. También relacionada con esta área, la geomática es la disciplina que se ocupa de la obtención, almacenamiento, análisis y explotación de la información geográfica [3].

Los drones se han convertido en una gran herramienta en el campo de la fotogrametría. Gracias al vuelo autónomo controlado, permiten capturar fotografías de la escena de forma sencilla. Además, son capaces de portar otras herramientas o sensores para realizar estudios más profundos sobre ella. Se hablará más sobre el uso de drones en este ámbito en el apartado 1.2.4 y, de forma más amplia, en el capítulo 2. En general, simplifican y abaratan los procesos necesarios, ya que proveen muchas ventajas y facilidades para la obtención de información geoespacial.

En este Trabajo de Fin de Grado, el objetivo general se resume en la gestión de información procedente de drones para su uso en esta área. Se aúnan la utilización y el desarrollo de una serie de herramientas con el objetivo de procesar extensos volúmenes de imágenes aéreas adquiridas mediante un dron, así como generar modelos 3D de información mediante un conjunto de técnicas que dan respuesta a las necesidades actuales del sector. En el apartado 1.3 se detallarán los objetivos de forma específica.

1.1. Motivación

Durante siglos, muchas personas han mirado al cielo y las aves mientras imaginaban todas las ventajas que podría dar el uso del aire como medio de transporte. Hoy día es posible, y millones de pasajeros, con sus correspondientes mascotas y equipaje, además de toneladas de mercancías comerciales, realizan vuelos de miles de kilómetros atravesando océanos. A pesar de ello, el ser humano como tal sigue sin disponer de herramientas para volar de forma individual y sencilla con el fin de realizar tareas o resolver problemas del día a día.

A expensas de que ese día llegue, el surgimiento de los drones y, específicamente, de los drones de uso civil, ha supuesto una verdadera revolución en la sociedad. Desde el sector de la agricultura hasta el cinematográfico, estas plataformas han demostrado su irrefutable utilidad, al dar en muchas ocasiones el control aéreo sencillo que siempre se ha deseado. No obstante, como en todo nacimiento de una nueva tecnología, el uso de drones ha supuesto y supone dar respuesta a una gran cantidad de retos. Poco a poco se van superando, pero se necesita tiempo e investigación.

El área de la fotogrametría aérea con drones no es una excepción. Aún existen multitud de cuestiones sin resolver; desde el aumento de la autonomía del dron hasta el procesamiento de datos. A pesar de ello, es un sector que se está desarrollando muy rápidamente y que cuenta con una gran proyección futura.

Por ello, en este trabajo se busca realizar un estudio sobre este nuevo paradigma de teledetección y reconstrucción aérea, surgido en los últimos años de la relación entre robótica, visión por ordenador y técnicas de geomática. Como objetivo final, se pretende demostrar la capacidad de este tipo de estudios y dar nuevas soluciones que faciliten y unifiquen las diversas tareas necesarias para realizar análisis de las características de superficies y terrenos. Soluciones que puedan dar respuesta a algunos de los desafíos presentes en la actualidad en este tipo de investigaciones.

1.2. Estado del arte

1.2.1. Primeros registros de uso de aeronaves no tripuladas y evolución durante el siglo XX.

Los vehículos aéreos no tripulados o drones modernos, tal como se conocen hoy en día, nacieron en el sector militar, de la misma forma que otros muchos grandes inventos de la humanidad [4].

Los primeros registros de uso de vehículos aéreos no tripulados se remontan al siglo XIX. Muchos concuerdan en marcar 1849 como el año en que se usó por primera vez el primer atisbo de estas plataformas. Aunque por aquella época aún no existía el término “dron”, ni se había producido el primer vuelo de los hermanos Wright, las fuerzas austríacas que sitiaban Venecia durante la Primera Guerra de la Independencia Italiana intentaron usar vehículos de este tipo para forzar la rendición y capturar la ciudad.

Vieron como en Venecia se solían dar vientos regulares desde el mar hacia la ciudad, por lo que idearon unos globos incendiarios no tripulados, cargados con explosivos. La idea era soltarlos desde los barcos austríacos, dejar que el viento los trasladara sobre la ciudad y que, gracias a un temporizador, se dejaran caer y las bombas detonasen al llegar al suelo.

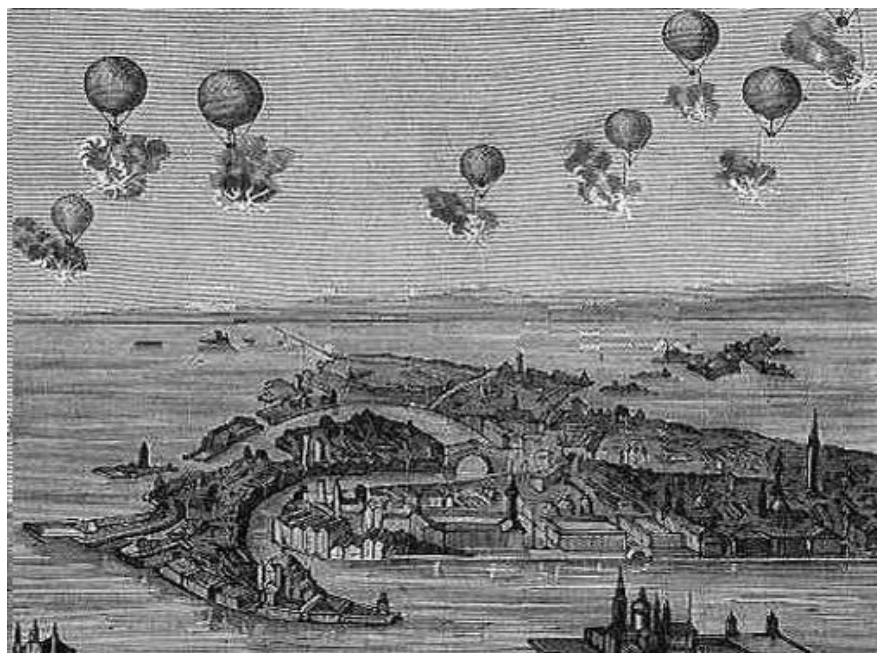


Ilustración 1.3 –Globos militares incendiarios austríacos cayendo sobre Venecia. Grabado en madera. Fuente: [5]

El resultado no fue del todo el esperado ya que, aunque sí hay indicios de que alguno de ellos logró detonar en la ciudad, la mayoría no lo hicieron e incluso algunos, debido a un cambio en las condiciones atmosféricas y la dirección del viento, se precipitaron sobre las propias filas austríacas. En todo caso, no causaron ningún daño digno de mención a ningún bando y el proyecto simplemente se abandonó. Evidentemente, estas aeronaves no tienen prácticamente ningún parecido a las que tenemos hoy en día, pero comenzaron a sentar las bases de este concepto.

Por su parte, también a finales de este siglo, en 1898, surgieron las primeras ideas de reconocimiento aéreo. Fue en la guerra Hispanoamericana, donde militares del ejército de los EEUU equiparon cámaras fotográficas a cometas para después elevarlas y tomar fotografías aéreas de los alrededores con las que seguir los movimientos del enemigo, formando mapas de situación.

1.2.2. Primera y Segunda Guerra Mundial

Pasaron varios años en los que surgieron inventos como el telégrafo o la radio, predecesores del que sería una de las grandes revoluciones de la robótica en general, el radiocontrol, que surge en la década de 1890. Este invento sentó las bases para que, una vez más con propósitos bélicos, durante la Primera y, sobre todo, Segunda Guerra Mundial, comenzaran a crearse las primeras aeronaves no tripuladas parecidas a las conocidas hoy día [6] [7]. Se pueden mencionar varios ejemplos:

En la Primera Guerra Mundial, el *Hewitt Sperry Automatic Airplane* fue uno de los primeros proyectos de aeronave no tripulada capaz de portar explosivos. La idea consistía en crear una aeronave no tripulada que despegara o fuera catapultada desde tierra, ascendiera a una altura predeterminada, realizara un recorrido preestablecido y, después de volar una distancia también fijada con anterioridad, soltara las bombas que portaba. Se realizaron varias pruebas y, a pesar de no acabar teniendo demasiado éxito, se considera a esta aeronave como la precursora de los torpedos.

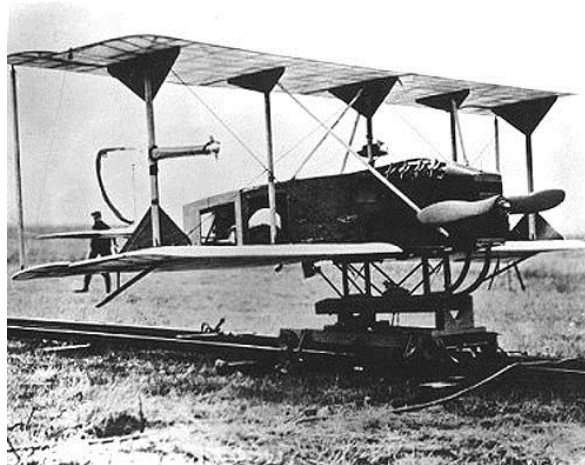


Ilustración 1.4 - Hewitt-Sperry Automatic Airplane, 1918. Fuente: [8]

Por su parte, en la Segunda Guerra Mundial se realizaron mucho mayores avances en este campo, debidos a la gran cantidad de pérdidas humanas de pilotos y de los propios aviones. En estos años, gracias a los avances tecnológicos, se comenzaron a construir de forma más avanzada aeronaves controladas a distancia mediante radiocontrol. Sirva como ejemplo el RP-5 estadounidense, más conocido como OQ-2 [9]. Aunque no se utilizaba directamente en combate, fue el primer dron producido en masa en los EEUU. Se usaron como objetivos a disparar en los entrenamientos de artilleros antiaéreos.



Ilustración 1.5 - Dron OQ-2. Fuente: [9]

1.2.3. Guerra Fría y finales de siglo

Durante el período de tiempo de la Guerra Fría hasta finales de siglo, varios países continuaron programas militares de mejora y desarrollo de estas aeronaves, especialmente, EEUU. El país norteamericano les dio un gran uso en la Guerra de Vietnam como armas y como herramientas de espionaje [6]. Tanto es así que el dron *Ryan Firebee* [10], contando todas sus versiones, continúa siendo hoy día uno de los

drones de vigilancia que más se han usado, a pesar de que inicialmente se utilizaba únicamente como objetivo de entrenamiento.

Todo esto llevó a que, a finales de siglo, ya existían drones como el MQ-Predator [11], capaces de memorizar rutas y objetivos sin necesidad de radiocontrol, sistema de control de tierra y conexión vía satélite, mientras contaban con un completo sistema de armas y que han servido en conflictos como Bosnia, Serbia, Afganistán, Yemen e Iraq. Pero no únicamente eso, sino que también comenzaron a verse las posibles ventajas de utilizar drones en otros sectores distintos al militar. Muestra de ello que en Japón comenzaron a utilizarse en tareas de fumigación de cultivos [1]. Antecedentes de la explosión comercial de los drones de uso civil, tema del que se hablará en el capítulo 2.

1.2.4. Uso de drones para estudios en los campos de la fotogrametría y geomática

Este trabajo se centra en la gestión de información procedente de drones en forma de fotografías de terrenos, que se usan para analizar y estudiar las características de la superficie captada. Este paradigma de teledetección y reconstrucción aérea ha surgido en los últimos años de la relación entre robótica, visión por ordenador y técnicas de geomática, y satisface actualmente muchas de las necesidades de buena parte de la industria para obtención de información geoespacial, contando además con una gran proyección futura.

Que los drones sean una buena herramienta para este tipo de estudios, no significa que sean la única forma de estudiar superficies de terreno. En el campo del análisis geográfico, en el pasado se podían obtener datos estudiando el terreno desde su propia superficie, a pie, sin ninguna imagen aérea, pero de forma mucho más complicada y menos eficiente. Es solo desde unas décadas atrás que se ha podido comenzar a emplear utensilios tales como aeronaves o satélites, así como nuevas técnicas y procesos relacionados con ellos que han sido añadidos a las herramientas de los geógrafos y geólogos [12]. La cuestión es que el uso de imágenes aéreas ha abierto un mundo de posibilidades en este tipo de estudios.

Sin embargo, el uso de aeronaves como avionetas o la recolección de imágenes vía satélite, a menudo requieren de la colaboración institucional o comercial de

terceros. En cualquier caso, el manejo de aeronaves tradicionales requiere una mucho mayor especialización profesional a medida que el sector progresa. Con la explosión durante este siglo del uso de drones fuera de la industria militar, esto deja de ser un problema. Estas plataformas pueden ser utilizados por prácticamente cualquier usuario una vez superada la burocracia requerida, que no suele ser demasiada. Con ello, además, se consigue una información y conjunto de datos más personalizado y ajustado a cada problema individual, mientras también resuelven desafíos como la adquisición de datos de terrenos en superficies remotas o de alta montaña, si las condiciones son favorables, donde no es seguro o es directamente imposible utilizar aeronaves tradicionales.

Muchas de las posibilidades que ofrecen los drones para estos estudios ya están disponibles, pero al igual que en el caso de otras tecnologías, de seguro se irán obteniendo más a lo largo del tiempo mediante el uso práctico. Por un lado, facilitan en gran medida las tareas requeridas para la investigación, abaratando también de forma clara los costes, pero también ofrecen una gran cantidad de nuevas alternativas si se relacionan con los nuevos avances en microsensores [12] y sensores multiespectrales. Gracias a ellos, por ejemplo, es posible captar longitudes de onda fuera del espectro visible, datos térmicos, etc. Información que permite interpretar y estudiar la escena de forma más profunda.

También, el que los drones sean una buena herramienta no significa que sean la única necesaria. Dan grandes ventajas realizando algunas partes del trabajo, pero un estudio va mucho más allá de la habilidad de los sistemas para adquirir datos sobre el espacio. Los drones se utilizan para proveer a investigadores y operadores una gran cantidad de información, con lo cual finalmente se realiza un análisis mayor, más rápido y más barato [12].

La base en la que se suele sustentar la fotogrametría en estos estudios de terreno es en la técnica “Structure from Motion” (SfM por sus siglas). Este es un método de bajo coste y efectivo que se basa en principios parecidos a los de la visión estéreo. Se entrará más en profundidad en estos algoritmos en el capítulo 5, pero con ellos se es capaz de obtener un modelo 3D a partir de una serie de imágenes. Todo ello sin la necesidad de tener ninguna información sobre la geometría de la escena, posiciones de las cámaras u orientación [13]. Todo se resuelve de forma automática,

determinando la localización 3D de características coincidentes entre las fotografías. Gracias a ello, únicamente con las fotografías del dron procesadas por herramientas que usen estas técnicas, se es capaz de realizar un estudio de calidad. Como ejemplo de aplicaciones o programas comerciales especializados en fotogrametría a partir de imágenes de drones, se pueden mencionar Agisoft PhotoScan, con una versión estándar y otra profesional; Pix4Dmapper, DroneDeploy o Drone2Map. Además, existen también alternativas gratuitas como DroneMapper RAPID.

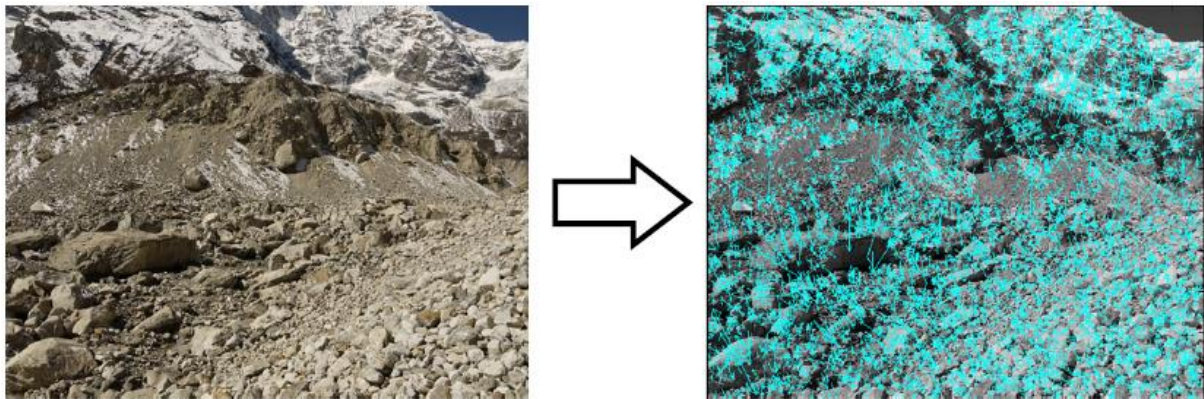


Ilustración 1.6 - Muestra de una de las fases de los algoritmos SfM. Se descompone una imagen dada (izquierda) en una serie de descriptores claves (derecha). Fuente: [13]

Los algoritmos SfM no solo son usados para estudios de terrenos, sino que también permiten y son utilizados para escaneado o mapeado 3D de monumentos, esculturas, edificios u objetos, por lo que pueden entrar en campos como la recuperación de patrimonio o medicina, en la que, entre otras cosas, se puede combinar con la impresión 3D para obtener modelos tridimensionales que pueden ser útiles para realizar prótesis. Incluso se están comenzando a utilizar en la industria de los videojuegos, con el fin de obtener modelos 3D de objetos cotidianos, áreas, etc. a partir de unas cuantas fotografías.

Una técnica que también resuelve el problema de la localización, mapeado y reconstrucción en tres dimensiones del entorno, es **SLAM** (Simultaneous Localization and Mapping). Ambas comparten algunos principios básicos, pero la principal diferencia con las técnicas SfM es que las técnicas SLAM se ejecutan en tiempo real y pueden usar información de otros sensores para ello. No obstante, también existen variantes de SLAM, como **vSLAM** (Visual SLAM), que usan únicamente imágenes que se van incorporando de forma secuencial al procesamiento.

Que este tipo de tecnología pueda ejecutarse en tiempo real tiene un precio en cuanto a calidad. SfM no se rige por este requisito, por lo que sus resultados son más pulidos. Sin embargo, en ciertos tipos de problemas, la ejecución en tiempo real puede ser indispensable. Por ejemplo, una de las mayores aplicaciones de SLAM es el guiado automático de robots o vehículos no tripulados en entornos desconocidos, un tema muy presente hoy día.

También existen otras alternativas a la fotogrametría, como emplear directamente algún sensor que pueda detectar la profundidad. Por ejemplo, la técnica **LIDAR** (Laser Imaging Detection and Ranging) consiste en utilizar un emisor láser que emite un haz pulsado y, midiendo el tiempo de retraso entre la emisión del pulso y la detección de la señal reflejada, es capaz de determinar la distancia [14]. No es una técnica mejor, pero tampoco peor, sino que, dependiendo del problema concreto, quizás una provea más ventajas que la otra. En todo caso, en este TFG no se cuenta ni se trabajará con este tipo de tecnología, sino únicamente con las técnicas de fotogrametría aplicadas sobre las fotografías captadas. Además, no se buscan soluciones en tiempo real, sino análisis lo más precisos posible, por lo que las técnicas de fotogrametría a usar se basan en SfM.

1.3. Objetivos

Bajo el contexto descrito, este trabajo se centra en la gestión de datos en forma de imágenes aéreas procedentes de drones, tomadas en vuelos controlados. El principal objetivo de este TFG consiste en analizar y facilitar ciertas tareas necesarias en la etapa de procesamiento de imágenes procedentes de drones para el estudio en el campo de la fotogrametría y geomática de características de terrenos o superficies, así como demostrar las capacidades de esta área.

Además, puesto que se trata de un proyecto de innovación y desarrollo, a medida que se avance en el trabajo y se vayan cumpliendo los objetivos, enunciados a continuación, se irán tomando y justificando ciertas decisiones en cuanto a qué caminos se han seguido y qué otros objetivos se podrían afrontar.

Las tareas iniciales que se buscan realizar y facilitar se pueden dividir en tres fases:

- **Selección de imágenes.** Organizando las imágenes tomadas y eliminando aquellas de las que se obtenga constancia de que no son necesarias o no aportan ningún tipo de información.
- **Reconstrucción virtual del terreno.** En base a las imágenes, aplicar distintas técnicas de fotogrametría y algoritmos sobre ellas para obtener las reconstrucciones virtuales del terreno en forma de nubes de puntos, malla de alambre y modelo tridimensional.
- **Análisis de la superficie.**
 - En primer lugar, en base a la reconstrucción, obtener características como modelos de la superficie estudiada, ortomosaico y textura.
 - A partir de la reconstrucción y haciendo también un análisis de las imágenes, extraer ciertas características interesantes del terreno. Para este análisis, se utilizarán las imágenes provenientes de un sensor multiespectral acoplado al dron que toma imágenes en distintas bandas espectrales.

Como se ha mencionado, cada fase se intentará automatizar lo máximo posible para simplificar las tareas a un posible usuario que necesite realizarlas.

Además, con el fin de unificar los resultados obtenidos y conocimientos del estudio realizado, se propuso como objetivo final la **realización de un prototipo**. Este representaría una aplicación con las siguientes características:

- Facilitará la visualización de las imágenes aéreas utilizadas para el proyecto.
- Será capaz de también visualizar los resultados obtenidos, de forma que el usuario pueda navegar fácilmente sobre ellos e interactuar con la cámara virtual para observarlos desde distintos ángulos.
- Mostrará detalles relativos al vuelo realizado por el dron para la captación de imágenes usadas en ese proyecto específico.

Dependiendo de la complejidad de la tarea que se vaya a emprender, se hará un análisis de las herramientas a utilizar para ello, así como el paso a paso de los procedimientos que se llevan a cabo -también internamente, si procede- durante su ejecución y para la obtención del resultado deseado.

1.4. Planificación

Seguidamente, se muestra la planificación del proyecto para el cumplimiento de los objetivos enunciados. Se hará una división de las tareas a realizar y, a cada una de ellas, se le asignará el tiempo estimado para completarlas. De esta forma, se obtendrá la planificación temporal del trabajo.

1.4.1. Identificación de tareas

Las distintas tareas por realizar para cumplir los objetivos iniciales y completar el proyecto se dividen en las siguientes:

- Estudio de soluciones actuales y estado del arte.
 - Investigación sobre el ámbito de los drones.
 - Investigación sobre la fotogrametría aérea y geomática.
 - Estudio del área de fotogrametría aérea con drones.
- Análisis.
 - Análisis de las necesidades para la clasificación y selección de imágenes.
 - Estudio del equipamiento utilizado.
 - Análisis del proceso de reconstrucción.
 - Análisis de requisitos del prototipo.
 - Elección de herramientas a utilizar.
- Desarrollo.
 - Preparación de las herramientas y entornos de desarrollo.
 - Desarrollo de scripts de clasificación y selección de imágenes.
 - Desarrollo de los scripts necesarios en la reconstrucción.
 - Reconstrucción del primer terreno.
 - Nube de puntos dispersa.
 - Nube de puntos densa.
 - Modelo 3D de la superficie.

- Otras características.
- Reconstrucción del segundo terreno.
 - Nube de puntos dispersa.
 - Nube de puntos densa.
 - Modelo 3D de la superficie.
 - Otras características.
- Estudio de terreno con las imágenes multiespectrales.
- Realización del prototipo.
- Redacción de la memoria del proyecto.

1.4.2. Planificación temporal

Atendiendo a la guía docente, según la carga de créditos del Trabajo de Fin de Grado, el tiempo de realización es de 300 horas. A razón de 3 horas diarias y 5 días laborales a la semana, resulta en un total de 100 días, o 20 semanas de trabajo. Teniendo esto en cuenta, así como los conocimientos previos del alumno, se muestra seguidamente una tabla con el tiempo estimado para cada tarea:

Nombre de la tarea	Días	Horas
Estudio de soluciones actuales y estado del arte		
Investigación sobre el ámbito de los drones	3	9
Investigación sobre la geometría aérea y geomática	3	9
Estudio del área de fotogrametría aérea con drones	4	12
Análisis		
Análisis de las necesidades para la clasificación y selección de imágenes	4	12
Estudio del equipamiento utilizado	6	18
Análisis del proceso de reconstrucción	10	30
Análisis de requisitos del prototipo	2	6
Elección de herramientas a utilizar	3	9
Desarrollo		
Preparación de las herramientas y entornos de desarrollo	2	6
Desarrollo de scripts de clasificación y selección de imágenes	3	9
Desarrollo de scripts necesarios en la reconstrucción	3	9
Reconstrucciones del primer terreno	12	36
Reconstrucciones del segundo terreno	15	45

Estudio de terreno con las imágenes multiespectrales	5	15
Realización del prototipo	10	30
Redacción de la memoria del proyecto	15	45
Total	100	300

Tabla 1.1 - División de tareas del proyecto. Fuente: Elaboración propia.

1.5. Coste del proyecto

En este apartado se descompone el coste total del proyecto, contando tanto los costes de hardware como de software, así como los propios costes humanos necesarios para el desarrollo del trabajo.

1.5.1. Costes hardware

A continuación, se desglosa el coste en hardware del proyecto. Se tendrán en cuenta los equipos y dispositivos utilizados. Se calculará el coste relativo, amortizando el coste total de los equipos con respecto a su vida útil, considerada de 5 años. Siendo el tiempo de desarrollo del proyecto unas 20 semanas, el coste amortizado corresponde a, aproximadamente, un 7.7% del total.

Los dispositivos utilizados para este proyecto se mencionan en profundidad en el capítulo 3. Estos son: dron Mavic Pro, sensor multiespectral Parrot Sequoia, teléfono móvil, ordenador en la universidad y ordenador personal. Las características principales del ordenador personal son procesador AMD FX-6300, 8 GB de memoria RAM y gráfica NVIDIA GeForce GT 740. Por su parte, el ordenador de la universidad cuenta con procesador Intel Core i5 750, 8 GB de memoria RAM y gráfica NVIDIA GeForce GT 220.

Dispositivo	Coste (€)	Coste amortizado (€)
Dron Mavic Pro	999	76,92
Sensor Parrot Sequoia	3200	246,4
Teléfono móvil	200	15,4
Ordenador universidad	600	46,2
Ordenador personal	800	61,6
Total	5799	446,52

Tabla 1.2 - Costes hardware. Fuente: Elaboración propia.

1.5.2. Costes software

El software utilizado en el proyecto es:

- Pix4Dcapture.
- Agisoft PhotoScan Professional Edition, contando con una licencia para uso educativo disponible al precio de 549\$, es decir, 473'73€, en pago único.
- Unity Personal. Unity es gratuito siempre y cuando los ingresos sean menores a 100.000\$ anuales o totales.

Software	Coste licencia (€)
Pix4Dcapture	0
Agisoft PhotoScan Professional Edition (Educational license)	473,73
Unity Personal	0
Total	473,73

Tabla 1.3 - Costes software. Fuente: Elaboración propia.

1.5.3. Costes humanos

Aunque el proyecto haya sido realizado en su totalidad por un solo alumno, se considerará que éste desempeña los diferentes roles necesarios en un proyecto real. Es decir: gestor de proyectos, analista y programador.

El salario medio de un gestor de proyectos en España, hoy en día, es aproximadamente de 31.751€ anuales [15]. El de un analista informático es de 26.847€ anuales [16]. Por último, el de un programador es de 17.470€ anuales [17]. La jornada laboral de cada trabajador es de 40 horas semanales durante los 12 meses del año. Así, el salario por hora correspondería a 16'53€ para el gestor de proyectos, 13'98€ para el analista informático y 9'10€ para el programador informático.

En este trabajo, el gestor de proyectos se encargará de, inicialmente la primera fase de "Estudio de soluciones actuales y estado del arte", estará involucrado en menor medida en la fase de análisis y, por último, la redacción de la memoria final. Por su parte, el analista se encargará de la propia fase de análisis en mayor medida. Por último, el programador llevará a cabo las tareas de desarrollo. Por lo tanto, una vez conocida, por una parte, la división de tareas y, por otra, el salario para cada trabajador, se obtiene el siguiente coste humano:

Concepto	Horas	Salario/hora (€)	Coste (€)	Coste total (+31% Seguridad Social)
Gestor de proyectos	90	16'53	1487'70	1948'89
Analista	60	13'98	838'80	1098'83
Programador	150	9'10	1365	1788'15
Total	300		3691'5	4835'87

Tabla 1.4 - Costes humanos. Fuente: Elaboración propia.

1.5.4. Coste total

Una vez calculados los costes hardware, software y humanos, el coste total del proyecto es únicamente la suma de ellos:

Concepto	Coste (€)
Costes hardware	446,52
Costes software	473,73
Coste humano	4835'37
Coste total proyecto	5756'12

Tabla 1.5 - Suma de costes del proyecto. Fuente: Elaboración propia.

1.6. Estructura de la memoria

En este apartado, a continuación, se expone un resumen de los siguientes capítulos de los cuales consta esta memoria:

En el capítulo 2, se realizará un estudio sobre el contexto actual del uso de drones y se comentarán aspectos importantes a conocer sobre ellos. Se hablará sobre sus áreas de aplicación y qué es lo que los convierte dentro de ellas en una mejor alternativa frente al uso de otras herramientas tradicionales. También se analizarán otras cuestiones que es indispensable conocer, como la ley por la que se rige el uso de estos vehículos.

Después, en el capítulo 3, se hablará sobre la primera fase del estudio de superficies mediante drones y fotogrametría: la adquisición de imágenes. Se mostrarán los primeros pasos y decisiones que habrá que considerar, el equipamiento

necesario y la forma correcta de adquirir las fotografías. Además, se presentará el equipamiento específico utilizado para la captura de imágenes en este trabajo y la metodología seguida para ello.

En el capítulo 4, se desglosarán los problemas encontrados en los conjuntos de imágenes completos que son capturados para este tipo de trabajos y se desarrollarán diversas soluciones para ellos. Estas se aplicarán sobre los conjuntos de fotografías utilizados en este trabajo.

A continuación, en el capítulo 5, se entra ya en el área de la fotogrametría en sí. Inicialmente, se hablará sobre las técnicas de fotogrametría que son generalmente aplicadas sobre las fotografías aéreas para los estudios en este campo. Después, se examinará el software utilizado en este trabajo para ello. Por último, se analizará la metodología seguida y los resultados obtenidos.

En el capítulo 6, se mostrarán algunas de las posibilidades que ofrece la captura y utilización de fotografías del terreno en distintas bandas espectrales. Específicamente, se usarán para realizar cálculos de índices de vegetación de la superficie.

Seguidamente, en el 7, se habla sobre el último objetivo que fue propuesto para este TFG: la realización de un prototipo software que unifique la visualización de todas las fotografías iniciales y todos los resultados obtenidos, así como detalles sobre el vuelo realizado. Se justifica la toma de esta decisión, se realiza un análisis de requisitos y se entra en detalles de su desarrollo.

Más tarde, en el capítulo 8 se exponen las conclusiones del presente TFG y, además, se ofrece una perspectiva sobre los posibles trabajos futuros.

Por último, en el capítulo 9, se realiza una descripción de los anexos adjuntos.

Capítulo 2.

Contexto actual en el uso de drones.

2. CONTEXTO ACTUAL EN EL USO DE DRONES

Antes de entrar en el trabajo, es importante conocer algunas cuestiones generales de estos vehículos aéreos no tripulados. Además, hay otros aspectos indispensables que se deben tener presentes a la hora de emplearlos como herramientas, como la ley por la que se rigen. En este capítulo se habla sobre ello, comenzando por un repaso general del estado actual del uso de estas plataformas:

La verdadera revolución de los drones de uso civil comenzó apenas una década atrás y, hoy en día, nos encontramos en medio de ella. De hecho, muchos países se han visto casi por sorpresa obligados a redactar una legislación que los incluya y regularice su uso, y que continuamente deben actualizar. En España, por ejemplo, de tener una regulación casi nula en 2014, en cuestión de seis meses se pasó a tener tres marcos normativos diferentes mediante un Decreto-Ley, que además fue actualizado en 2017 [18] [19].

Esta necesidad de regular su uso está provocada principalmente por los posibles problemas de privacidad y seguridad que pueden surgir con el uso de estas plataformas. Son herramientas tan versátiles y silenciosas que podrían ser utilizadas, equipando una cámara que realice fotografías, vídeos o grabación en tiempo real, para sobrevolar y captar imágenes de propiedades privadas ajenas tales como casas, empresas, fincas, etc., pudiendo violar así derechos como la intimidad personal y familiar. Pero no solo eso, incluso siendo usados en espacios públicos o siendo utilizados por empresas como herramienta de vigilancia o de grabación de eventos, estos están incurriendo en cuestiones tan críticas e importantes como la protección de datos de carácter personal [20]. Además, los drones ya han sido utilizados en otras actividades fraudulentas como sobrevolar centrales nucleares, interferir en comunicaciones o el narcotráfico y, teniendo en cuenta sus inicios, no es de extrañar que también puedan llegar a usarse como armas. Por motivos como estos, la regulación era y sigue siendo una cuestión necesaria. A continuación, se enumeran diversos aspectos de la regulación que será indispensable tener en cuenta si se desea utilizar drones en el ámbito profesional y, por supuesto, en el de este trabajo.

2.1. Regulación actual

Las normas actuales sobre el uso de drones se rigen sobre el Real Decreto 1036/2017, de 15 de diciembre, por el que se regula la utilización civil de las aeronaves pilotadas por control remoto, y se modifican por el Real Decreto 552/2014, de 27 de junio, por el que se desarrolla el Reglamento del aire y disposiciones operativas comunes para los servicios y procedimientos de navegación aérea y el Real Decreto 57/2002, de 18 de enero, por el que se aprueba el Reglamento de Circulación Aérea [21].

Los aspectos más importantes son los siguientes [22] [21]:

- Si se desea volar sobre zonas urbanas y aglomeraciones de personas, se deben cumplir las siguientes condiciones:
 - La masa de la aeronave no debe superar los 10 kg.
 - El dron siempre debe estar al alcance visual del piloto.
 - La zona por volar deberá estar acordonada por la autoridad competente, o en todo caso, mantener una distancia horizontal mínima de 10 metros con respecto a edificios o estructuras y con otras personas ajenas a la operación.
 - La aeronave debe contar con un sistema de limitación de energía del impacto, como paracaídas o airbag.
 - Se debe contar con la autorización de AESA.
- Si se desea realizar un vuelo nocturno, se debe contar con la autorización de AESA y la aeronave deberá incorporar dispositivos como luces que garanticen su visibilidad.
- Si se va a desempeñar una actividad profesional con drones, se debe obtener una licencia de piloto, acreditando unos conocimientos teóricos y prácticos. Además, se debe estar dado de alta como operador en AESA, tener seguro de responsabilidad civil y tener certificado médico en vigor.
- Podrán realizarse vuelos fuera del alcance visual del piloto siempre dentro del alcance directo de la emisión por radio de la estación de pilotaje remoto, cuando se cuente con sistemas certificados y autorizados por AESA que permitan detectar y evitar a otros usuarios del espacio aéreo. Como excepción a lo último, se pueden realizar vuelos más allá del

alcance visual del piloto cuando la aeronave sea de menos de 2 kg de peso, y el vuelo sea menor a 400 metros de altura.

- Cualquier aeronave no tripulada debe llevar una placa identificativa ignífuga.

Habrá que tener pleno conocimiento de ellos para realizar un cumplimiento correcto de la ley en cualquier tipo de trabajo con drones. Asimismo, algunos de ellos puedan ser considerados desventajas que habrá que sopesar si se quiere utilizar este tipo de vehículos como herramientas para fotogrametría aérea.

2.2. Ventajas que provee la utilización de drones

Desde que comenzaron las primeras operaciones con drones, el ecosistema tecnológico ha experimentado un gran avance y ha puesto a disposición del sector nuevas soluciones técnicas que están permitiendo el desarrollo de su gran potencial. Estas mejoras han dado lugar a una significativa miniaturización de equipos y un abaratamiento de los componentes, lo que se ha traducido en una mayor accesibilidad de las empresas para hacer frente al diseño de nuevos sistemas que, finalmente, han provocado su rápida difusión en el ámbito civil. Ejemplo de esto, es la reducción de en torno al 90% que han experimentado los costes de fabricación y venta de estas plataformas durante los últimos 5 años [1].

Algunas de las ventajas que da el uso de vehículos aéreos no tripulados se podrían numerar en [1]:

- Disminución de costes comparados a aeronaves tripuladas en cuanto al propio precio de la aeronave y costes operativos, como el combustible o el mantenimiento.
- Mejora de la seguridad en el ámbito laboral, ya que no exponen a un piloto, aunque realicen trabajos en entornos peligrosos.
- Bajo impacto ambiental.
- Ampliación de las posibles aplicaciones, ya que son muy versátiles y pueden realizar tareas que no pueden asumir las aeronaves convencionales.
- Bajo tiempo de formación.

Estas ventajas han hecho que, además de muchas personas que los usan como pasatiempo o en tareas privadas, muchas empresas los vean como herramientas que pueden hacerlas mejorar en competitividad.

2.3. Áreas de aplicación

Como se apuntaba, la evolución constante que están teniendo los diferentes elementos tecnológicos que conforman los drones, hace que cada vez haya en el mercado modelos con mejores prestaciones, que permiten plantear su utilización en nuevos campos y aplicaciones. El empleo de nuevos materiales, más ligeros y flexibles; el desarrollo de baterías de menor peso y con una mayor capacidad, unido a la miniaturización y abaratamiento de la electrónica, así como el desarrollo de nuevos sensores, hacen posibles plataformas cada vez más potentes y versátiles y con mayores capacidades para la captura y proceso de datos [1].

Las áreas donde están entrando o podrían entrar en escena son inmensamente amplias y van desde la agricultura hasta el cine para filmación aérea de forma controlada y versátil. Todo ello pasando por el sector de la construcción, el entretenimiento, control del tráfico, mantenimiento y revisión de estructuras, control medioambiental de plantas y animales o vertidos ilegales, envío de paquetes, transporte de personas, etc.

Por ejemplo, en el campo de la agricultura de precisión se suele monitorizar el estado de las plantas para detectar áreas con enfermedades, deficiencia de nutrientes, plantas invasivas, patógenos o sequías. Con ello, la finalidad es aplicar fertilizantes, pesticidas, herbicidas y riegos a cada cultivo y región específica en base a sus necesidades, en lugar de al campo completo, mejorando el producto final y ahorrando recursos. Los drones son usados como herramienta de bajo coste para captar las imágenes aéreas de los cultivos, que más tarde se usan para analizar todo este tipo de cuestiones. Incluso ya se están realizando estudios en los que los propios drones son quienes, de forma automática y en tiempo real, detectan los posibles problemas o situaciones específicas en el cultivo y reaccionan tomando sus propias decisiones de forma inteligente, por ejemplo, rociando aerosoles en esas zonas o activando los sistemas de regado [23], entrando poco a poco en juego la Inteligencia Artificial.

Otra muestra concreta de área actual de aplicación de estas plataformas es la supervisión de construcciones y obra civil. En primer lugar, pueden ser usados para la revisión de puntos de la obra de difícil acceso o altura elevada, con el fin de comprobar que el estado de la estructura es el correcto y no existen defectos, o realizar ciertos tipos de comprobaciones con distintos sensores para asegurarse de que no existen filtraciones o vicios ocultos. Todo ello sin afrontar ningún riesgo humano y con un mínimo impacto. También puede ser usado para realizar un seguimiento en el tiempo del estado de la obra y ver su progreso escalonado a lo largo de las semanas o días, constatando que los plazos se van cumpliendo. Incluso pueden ser usados para tareas con no tanta utilidad en cuanto a la propia obra, pero también valiosas, como realizar tomas aéreas de calidad para vídeos corporativos de índole comercial o para mostrar al cliente [24].

2.4. Clasificación general atendiendo a su arquitectura

La gran cantidad de modelos existentes y de variadas tareas que pueden ejecutar, hace que los drones se puedan clasificar atendiendo a muchos factores. En este apartado se agrupan en base a los más comunes y en cuanto a arquitectura. En cada categoría, además, se mencionan algunos subtipos [1].

- **Rotor único.** Cuentan con un único rotor en forma de hélice.
 - Convencional.
 - Coaxial. Dos hélices, cada una girando en un sentido.
 - Nano.

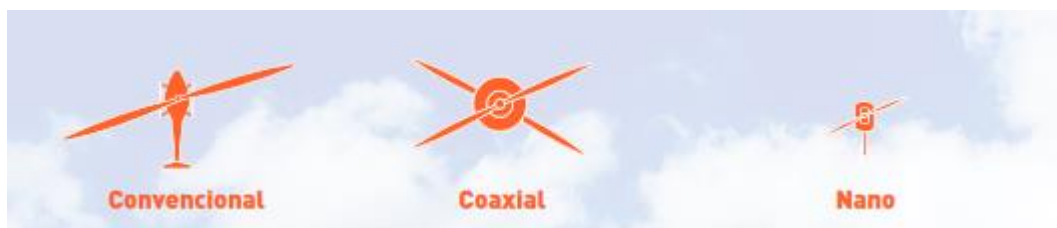


Ilustración 2.1 - Algunas clases de drones de rotor único. Fuente: [1]

No son los más comunes, pero son usados generalmente cuando se requiere una buena resistencia y cargar con un cierto peso. En el caso del Nano, cubre necesidades específicas en las que se requiere un muy pequeño tamaño.

- **Multirrotor.**
 - Tricóptero.
 - Cuadricóptero.
 - Hexacóptero.
 - Etc.



Ilustración 2.2 - Algunas clases de drones multirrotor. Fuente: [1]

Gracias a su buena estabilidad y controlabilidad, son los más comunes en los campos de fotografía y filmación aérea.

- **Híbrido.**
 - VTOL ala fija. Las hélices permiten el despegue vertical mientras cuenta también con alas fijas.



Ilustración 2.3 - Ilustración de dron híbrido VTOL de ala fija. Fuente: [1]

Comparte las ventajas de los drones de ala fija con la seguridad del despegue y aterrizaje vertical, lo que los lleva a ser usados en áreas muy similares a las de los drones de ala fija en situaciones con poco espacio para despegue y aterrizaje.

- **Ala fija** [25].
 - Dron de pequeña envergadura.
 - MALE. Altitud media largo alcance.
 - HALE. Altitud elevada largo alcance.
 - Ala alta. El ala se ubica en la parte superior del fuselaje. Mayor estabilidad.

- Ala media. El ala se sitúa en la parte media del fuselaje. Equilibrio entre estabilidad y maniobrabilidad.
- Ala baja. El ala se sitúa en la parte inferior del fuselaje. Mayor maniobrabilidad.



Ilustración 2.4 - Algunas clases de drones de ala fija. Fuente: [1]

Cuentan con una gran autonomía, lo que los convierte en las herramientas idóneas cuando se requiere abarcar grandes extensiones de terreno. Son usados en áreas como la vigilancia o cartografía.

Para los trabajos de fotogrametría, generalmente la opción escogida suelen ser los drones multirrotor, gracias a su gran estabilidad aérea y maniobrabilidad para cubrir la superficie específica deseada, además de la facilidad de aterrizaje y despegue. Sin embargo, por ejemplo, si se desea abarcar la superficie de un gran terreno, los drones de ala fija o, como mejor alternativa, los híbridos VTOL de ala fija, buenos en aterrizaje y despegue, suelen desempeñarse mejor, debido a su mucha mayor autonomía y capacidad de cargar peso.

Capítulo 3.

Adquisición de imágenes.

3. ADQUISICIÓN DE IMÁGENES

En este capítulo se entra ya en el propio trabajo, hablando sobre los primeros pasos y decisiones que hay que tomar para el estudio mediante drones y fotogrametría, consistentes en la adquisición de las imágenes del objeto de estudio. Primeramente, se hará una introducción en cuanto a las herramientas necesarias y cosas a tener en cuenta para realizar esta tarea de forma correcta. Más tarde se hablará sobre el equipamiento utilizado para capturar las fotografías de muestra para este TFG y, por último, la metodología seguida para ello.

3.1. Introducción y primeras consideraciones

La adquisición de imágenes es la primera fase crítica del análisis geográfico con drones. De ella saldrá el conjunto de fotografías que servirá como entrada a los algoritmos de reconstrucción y análisis y que conforman la totalidad del conjunto de datos. Es crucial que las fotografías se tomen de forma correcta para obtener los resultados esperados a medida que se trabaja con ellas.

Lo primero que hay que realizar es la definición del plan de vuelo. El plan de vuelo es un informe usado en la aviación tradicional en el que se reflejan todos los datos referentes a un vuelo. Entre otros, incluye el lugar de salida, destino, altitud, velocidad de crucero y todas las coordenadas geográficas que definen la trayectoria de la aeronave [26]. Traspasándolo al ámbito de los drones y a la fotogrametría aérea, se debe realizar este informe donde se incluya el lugar donde se pretende realizar el vuelo, un pequeño estudio sobre la superficie a abarcar, forma de hacerlo, fecha elegida, factores meteorológicos previstos para ese día, determinación de si son favorables, revisión del equipamiento necesario para ello, factores que se buscan analizar, etc.

El propio dron, por su parte, como necesidades básicas, debe ser capaz de equipar cámara, alcanzar la altura necesaria y desplazarse de forma correcta controlado por el operario o de forma automática según un recorrido precalculado y programado mientras se realizan las fotografías.

En cuanto al planteamiento del vuelo, se deben tener en cuenta varios factores. En primer lugar, por supuesto, se debe tener en consideración que las fotografías

deben abarcar la totalidad de la superficie de interés, pero deben hacerlo de la forma correcta. Los algoritmos con los que se trabajan para la reconstrucción 3D a partir de fotografías, como se mencionó anteriormente, funcionan con la premisa de que las imágenes deben tener un cierto grado de superposición entre ellas -se explica detalladamente en el capítulo 5.1-. Es un requisito crucial que no se puede evitar y por el que se deberán tener muy en cuenta tanto la velocidad de desplazamiento del dron, como el intervalo de tiempo que transcurre entre que se toman dos fotografías distintas.

Asimismo, la forma correcta de tomar las fotografías para una posterior generación de mapas 3D es manteniendo la cámara aérea paralela al terreno y desplazando únicamente el dron para tomar capturas distintas de la superficie, en lugar de mantener el dron inmóvil y rotar la cámara. En la Ilustración 3.1, se aclara lo dicho.

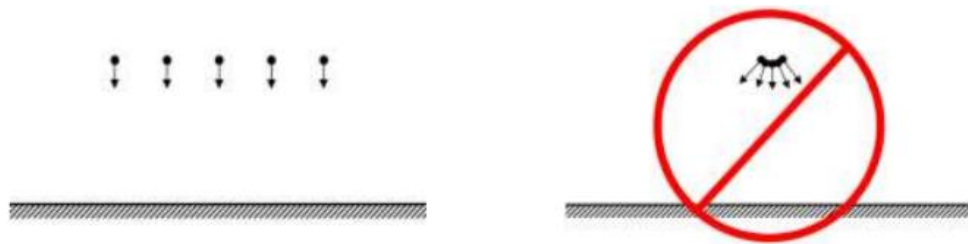


Ilustración 3.1 - Image Capture Tips: Equipment and Shooting Scenarios. Fuente: [27]

Además, los procedimientos necesarios para conseguir una reconstrucción del terreno en base a imágenes con técnicas SfM son bastante costosos en cuanto a tiempo y recursos se refiere. Dependiendo del conjunto de imágenes, pueden tardar varias horas o incluso días en ejecutarse en un ordenador común. Es por ello que, en general, se busca disminuir, dentro de lo posible, el conjunto de imágenes inicial. Este, en la mayoría de los casos, puede contener desde algunas decenas de fotografías hasta varios centenares de ellas. Se busca utilizar únicamente aquellas que aporten realmente una información del terreno valiosa y que pueda ser aprovechada.

En el ámbito de este trabajo, a continuación, teniendo en cuenta la forma general descrita en la que se capturan las imágenes, se indaga más en el dron específico utilizado para capturar el conjunto de datos con el que se trabajará, las características

de la cámara equipada, así como configuración y detalles de la metodología necesaria a seguir para la adquisición de las imágenes.

3.2. Dron y equipamiento utilizado

En este apartado se exponen las herramientas usadas para las capturas de imágenes de terreno tomadas como muestra para este trabajo, así como sus características. Se comienza por el dron utilizado para después indagar en la cámara y, después, en la aplicación controladora. Por último, se indica la configuración y, brevemente, metodología seguida.

3.2.1. Dron

El dron utilizado para captar el conjunto inicial de fotografías utilizado en este proyecto es un **Mavic Pro**, de la empresa china DJI [28], referente en la fabricación y desarrollo de drones.

Algunas de sus características más importantes son [29]:

- Arquitectura multirrotor, específicamente en forma de cuadricóptero.
- Tiempo de vuelo de unos 27 minutos.
- Cámara de 12 megapíxeles, capaz de grabar en 4K a 30 fps, con un estabilizador en tres ejes automático.
- Velocidad de hasta 65 Km/h.
- Rango de control de 7 Km.
- Sensores visuales frontales e inferiores con los que es capaz de mantenerse estático en interiores y lugares sin GPS, así como detectar obstáculos hasta a 15 metros de distancia y sortearlos.
- Soporta vientos de hasta 38 km/h.
- Capacidad de plegado y transporte en una bolsa pequeña.

Los principales atractivos de este dron para ser utilizado en trabajos de fotogrametría son su gran versatilidad en vuelos de propósito específicos como estos, la capacidad de volver al punto de despegue en caso de pérdida de contacto o batería baja, lo que da mucha seguridad; la capacidad de precalcular el recorrido y realizarlo de forma automática, su gran estabilidad que permite realizar fotografías nítidas

durante el vuelo y los ya mencionados sensores para evitar colisiones y que permiten volar a baja altura entre posibles obstáculos. Todo ello además de su pequeño tamaño que facilita en gran medida el transporte.



Ilustración 3.2 - Mavic Pro. Fuente: [29]

3.2.2. Sensor equipado

Para este trabajo, se cuenta con un sensor multiespectral **Parrot Sequoia** [30]. Algunas de sus características son:

- Capacidad de realizar fotografías en cuatro bandas espectrales distintas (verde, rojo, borde rojo e infrarrojo cercano).
- Cámara RGB de 16 megapíxeles.
- Sensor solar.
- Integración de módulo GPS que usa para, dependiendo de la situación, velocidad de vuelo y altitud, adaptar su velocidad de captura de fotografías, si fuera necesario.
- Almacenamiento de 64 GB, ampliables mediante tarjeta SD.

Con estas características, la cámara Parrot Sequoia se especializa en el sector de la agricultura de precisión, que es donde encuentra su mayor lugar en el mercado. Ayuda a los agricultores u operarios a predecir áreas que necesitan regarse más, enfermedades en las plantaciones, el rendimiento de los cultivos o zonas que necesitan una mayor acción con pesticidas. Sin embargo, también se usa en otros muchos ámbitos.



Ilustración 3.3 - Parrot Sequoia. Fuente: [30]

La configuración de esta cámara puede realizarse a través de algún software de planificación de vuelo compatible o mediante Wi-Fi vía interfaz HTML. Con la opción Wi-Fi, en la pantalla inicial a la que se accede introduciendo en el navegador la dirección IP que se indica en el manual, se permite configurar el modo de captura y ciertos detalles de los sensores: los que se quieren habilitar durante el vuelo (verde, rojo, borde rojo, infrarrojo cercano o RGB), la resolución de imagen y el *bit depth*, el número de bits usados para representar el color. También muestra valores relacionados con el estado de la cámara como la temperatura, el GPS, los sensores, etc. [31].

El modo de captura al que se alude en el párrafo anterior representa las distintas formas posibles que tiene la cámara para realizar las fotografías:

- *Single*: Toma las fotografías una a una.
- *Time-lapse*: Toma ráfagas de fotografías en intervalos regulares especificados en segundos.
- *GPS*: Toma ráfagas de fotografías a cada cierta distancia especificada en metros.

Es importante mencionar que, para que comience a capturar fotografías, es necesario pulsar manualmente un botón de captura. No lo hace a partir de una orden del dron. Y lo mismo ocurre en el caso de detenerla al terminar el trabajo; hay que pulsar el botón de parar. Esto influirá directamente en el filtrado de fotografías, que se desarrolla durante el capítulo 4 de este trabajo.

3.2.3. Aplicación para la planificación del vuelo

La aplicación móvil utilizada en este caso es Pix4Dcapture [32], disponible en los sistemas operativos Android e iOS. El dron utilizado, el Mavic Pro, tiene soporte para ella, así como a otras herramientas de Pix4D.

Pix4D es una empresa especializada en software relacionado con drones y análisis geográfico o fotogramétrico de imágenes capturadas por drones. Cuenta con aplicaciones como la mencionada Pix4Dcapture para el control del dron y la planificación del vuelo, o el ya mencionado Pix4DMapper [33], un software profesional de fotogrametría a partir de puramente imágenes capturadas por dron.

Pix4Dcapture permite establecer el área que se pretende cubrir con el dron, así como la ruta seguida para ello.

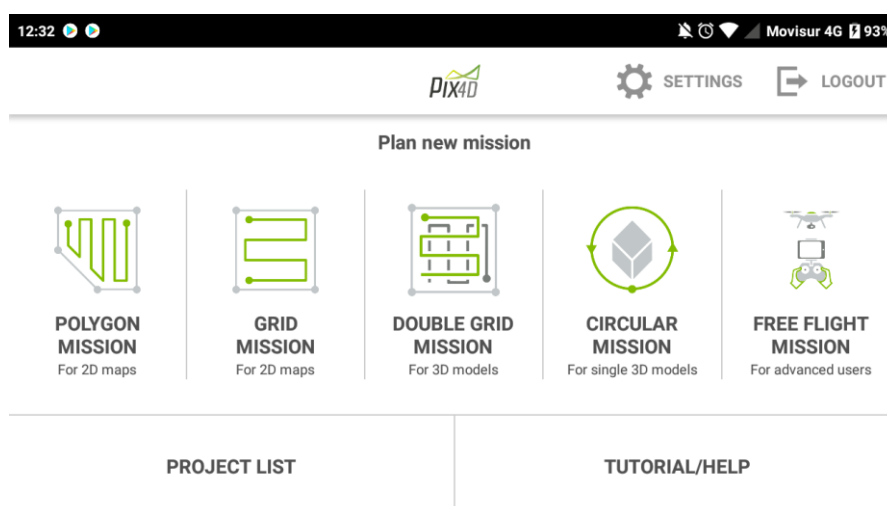


Ilustración 3.4 - Pantalla de elección del tipo de misión en Pix4Dcapture. Fuente: Elaboración propia.

Como se puede ver en la Ilustración 3.4, inicialmente la aplicación permite elegir la forma de abarcar el área. En definitiva, la forma general del recorrido que seguirá el dron para capturar las imágenes. Las dos primeras comenzando por la izquierda se usan mayoritariamente para mapas 2D; la tercera para modelos 3D, realizando un recorrido que se podría decir doble; la cuarta para modelos 3D aislados, como podría ser una escultura, y la última permite el vuelo libre, controlado remotamente por el usuario.

Después de escoger uno de estos modos, a excepción del último, se muestra un mapa con la posición geográfica actual del usuario y en el que se pueden establecer la posición y las dimensiones de la superficie a cubrir. Es posible intercambiar vista satélite con mapa 2D y también pueden configurarse otros parámetros. Se muestra un ejemplo en las siguientes imágenes:



Ilustración 3.5 - Pantalla para configuración del área a abarcar con vista en modo satélite. Fuente: Elaboración propia.



Ilustración 3.6 - Pantalla para configuración del área a abarcar con vista en modo mapa 2D y detalle de los ajustes. Fuente: Elaboración propia.

En la Ilustración 3.6 pueden verse algunos ajustes que puede establecer el usuario. Puede fijar una velocidad relativa de desplazamiento del dron más lenta o más rápida, el ángulo de la cámara del dron y la cantidad relativa de superposición necesaria. Es importante señalar que los ajustes relativos a la cámara no afectan a la cámara multiespectral utilizada que cuenta con los suyos propios.

A parte de ellos, en la parte izquierda de la pantalla, se permite establecer la altura a la que el dron realizará la misión. Es importante conocer que esta altura no es en todo momento relativa al terreno, sino únicamente relativa al punto de despegue, lo que influye en el plan de vuelo.

3.3. Metodología seguida y configuración utilizada

Tras haber realizado una buena definición del plan de vuelo y, suponiendo que las condiciones son favorables y que el dron está preparado para el trabajo, los ajustes que hay que establecer son los relativos a la cámara Parrot Sequoia y a la aplicación Pix4Dcapture.

Comenzando por la cámara multiespectral, una vez ajustada al dron, se estableció que capturara fotografías tanto en RGB como en las cuatro bandas espectrales de las que dispone. Además, el modo de captura se estableció en *time lapse* en intervalos de unos segundos.

Por otro lado, para poder usar la aplicación Pix4Dcapture, en primer lugar, hay que descargarla. Se encuentra disponible en la Play Store en el caso de usar un dispositivo Android o desde la App Store en caso de un dispositivo iOS.

Una vez descargada en nuestro móvil o Tablet, hay que conectar el dispositivo al dron. El Mavic Pro, en su mando, dispone de un conector micro USB y sujeciones para conectar la Tablet o móvil. En el momento en que la aplicación esté iniciada, aparece una primera pantalla donde se pide seleccionar el dron utilizado. En este caso, se selecciona Mavic Pro. Se pide además que iniciemos sesión con una cuenta de Pix4D. Si no se dispone de una, habrá que crearla introduciendo los datos personales que se pidan.

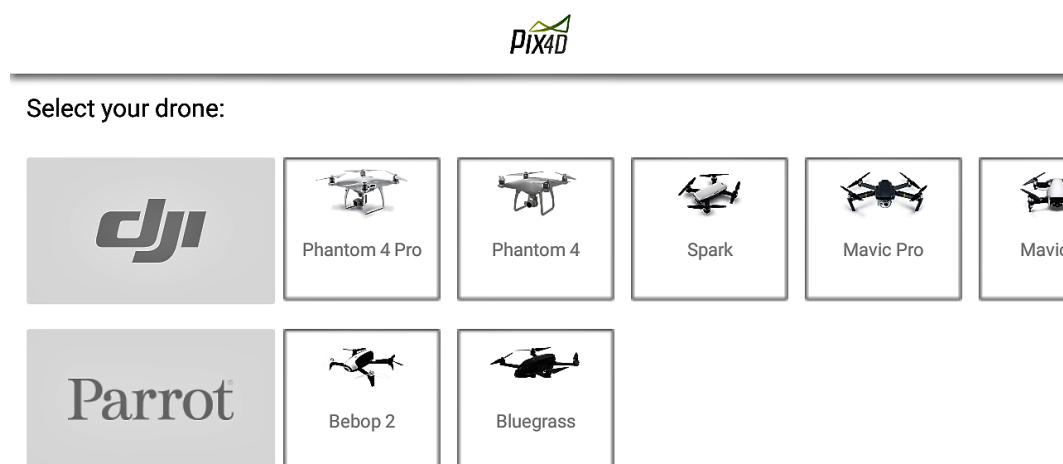


Ilustración 3.7 - Pantalla de Pix4Dcapture en la que se pide seleccionar el dron utilizado de entre todos los compatibles. Fuente: Elaboración propia.

Cuando se completan estos pasos, ya se muestra la pantalla a la que se hace alusión en la Ilustración 3.4. Para la reconstrucción 3D de terrenos, el mejor tipo de misión es el *Double Grid Mission* en la que el dron realiza una doble pasada sobre el área para maximizar las posiciones desde las que se realizan fotografías, asegurándose de no dejar parte de la superficie sin abarcar o con demasiada poca información obtenida y de conseguir el nivel de solapamiento requerido para este tipo de trabajos.

Por último, se define el área que se quiere abarcar. Como se ve en la Ilustración 3.5 e Ilustración 3.6, se pueden escoger para ello dos vistas distintas. En sendas pantallas, como se indicó, también se establecen los parámetros del propio vuelo como son la altura relativa o la velocidad, así como otros relacionados con la cámara del dron que pueden ser menos útiles si, como en este caso, se utiliza otra cámara distinta. En cualquier caso, si se va a usar, siguiendo las instrucciones de la Ilustración 3.1, debe fijarse totalmente en vertical, así como una superposición u *overlap* de no menos del 70% si queremos obtener una reconstrucción de calidad.

Una vez completada toda la etapa de planificación, se puede poner en marcha el plan de vuelo y el sensor multiespectral. El dron iniciará el despegue automáticamente y se dirigirá al área donde, una vez sobre ella, comenzará a realizar el recorrido establecido, así como a realizar fotografías con su propia cámara que pueden visualizarse en tiempo real vía Wi-Fi. Al finalizar, volverá y aterrizará también

de forma automática, gracias a los sensores de los que dispone. Durante todo el proceso, el sensor multispectral también tomará las fotografías correspondientes. A pesar de que el recorrido es realizado de forma automática, el dron siempre responderá a las órdenes del mando radiocontrol. Por ello, lo ideal es que el piloto esté atento durante todo el vuelo por si necesitara realizar alguna corrección con el fin de evitar algún obstáculo o circunstancias imprevistas.

Cabe destacar que, para este tipo de actividades, todo el proceso de planificación y control del propio vuelo debe ser realizado por una persona con el carnet autorizado de piloto de drones. Así se realizó para la captación de los conjuntos de imágenes utilizadas en este trabajo. Fueron obtenidos por la operadora TIC-144 de la Universidad de Jaén, que incluye cinco pilotos homologados de drones.

Capítulo 4.

Clasificación y selección de imágenes.

4. CLASIFICACIÓN Y SELECCIÓN DE IMÁGENES

En este capítulo, una vez vista la metodología y el funcionamiento o comportamiento general del dron, así como de la cámara auxiliar, se estudian algunos de los problemas encontrados en la totalidad del conjunto de imágenes una vez son importadas al ordenador. A continuación, se examinan y desarrollan algunas posibles soluciones a ellos.

4.1. Problemas encontrados

Con el equipo usado y descrito y, también en general en este tipo de trabajos, la duración del vuelo del dron suele rondar los veinte minutos. Durante ese período, como se estableció inicialmente, el dron y el sensor Parrot Sequoia capturan imágenes cada cierto intervalo de tiempo. Este intervalo es de únicamente unos segundos, por lo que, al final, la totalidad de las imágenes suele llegar a sumar varias centenas, dependiendo del caso. Esto sumado a que, como se mostrará en capítulos posteriores, el procesamiento necesario para una reconstrucción 3D en base a imágenes es muy costoso, y el coste se agrava a más fotografías se utilicen, hace que aumente el valor de realizar algún tipo de proceso de selección de imágenes.

Además, como se mencionaba en el capítulo anterior cuando se describían las características de la cámara multiespectral, así como también en la metodología seguida, esta cámara comienza a capturar fotografías desde que el usuario pulsa el botón antes del despegue y continúa tomando fotos hasta que se completa el programa y el mismo aterrizaje, cuando el usuario pulsa el botón de parada. Estas fotos de despegue y aterrizaje no son útiles. Suelen ser fotos desenfocadas, borrosas y que, en cualquier caso, no tienen la altura suficiente para dar alguna información relevante para el proceso de reconstrucción del terreno e incluso pueden aportar información errónea. Todo ello además de aumentar el tiempo de procesamiento en las siguientes etapas, lo cual es un factor clave.

Por otra parte, el dron y la cámara multiespectral por sí solos no clasifican las imágenes para cada vuelo. Si se han realizado varios, ya sea en el mismo terreno o en distintos lugares, todas las imágenes del dron se encontrarán en una misma carpeta, y todas las imágenes de la cámara multiespectral, se encontrarán en otra misma carpeta distinta. Eso sí, de forma ordenada numéricamente: desde la primera

foto captada del primer vuelo, hasta la última del último vuelo. No obstante, es el usuario el que tiene que comprobar y clasificar de forma manual las imágenes que pertenecen a cada vuelo individual y, probablemente, a un terreno concreto, y hacerlo tanto para las imágenes del dron como para las de la cámara multiespectral.

Con el objetivo de resolver estos inconvenientes, se decidió realizar dos scripts. Uno de ellos para filtrar las imágenes y eliminar aquellas que pertenezcan a despegues y aterrizajes y, el segundo, para clasificar las imágenes por vuelo.

En los siguientes subapartados, se tocarán algunas cuestiones importantes para la realización de estos scripts, la teoría propia en la que se basan, así como diversas cuestiones propias de su desarrollo.

4.2. Exchangeable image file format (Exif)

Para conseguir realizar una clasificación de ciertos elementos según varios factores, será necesario poder acceder a estos últimos en cada sujeto individual. Es decir, si se quieren filtrar, como se ha comentado, imágenes según la altura a la que se realizaron, será necesario conocer la altura a la que se realizó cada imagen en particular.

Los metadatos, o **datos Exif** (por las siglas en inglés de “Exchangeable image file format”), son aquellos contenidos en fotografías realizadas por cámaras digitales. Éstos se encuentran incrustados en formatos de imagen tales como JPEG o TIFF. Precisamente estos son los formatos en los que se suelen encontrar las fotografías para estos tipos de trabajos, capturadas por drones. Por un lado, las fotografías RGB, por norma general, se toman en formato JPEG y, por otra parte, las multiespectrales en formato TIFF. En este sentido, las fotografías para este trabajo no son una excepción.

Los datos Exif recogen gran cantidad de información para cada imagen tales como su nombre, directorio, tamaño, formato, modelo de cámara con la que se capturó, altura y anchura en píxeles, fecha y hora a la que fue tomada, longitud focal, posición GPS, altura sobre el nivel del mar, nombre de la banda en la que se tomó, etc. En la Ilustración 4.1 se muestran algunos de los datos Exif contenidos en una imagen utilizada en este trabajo, para lo cual se ha usado el archivo ejecutable para

Windows de la librería ExifTool [34], capaz de mostrar todos los datos de manera sencilla: arrastrando y soltando únicamente la imagen sobre el ejecutable. Esta aplicación se puede descargar fácilmente desde la página oficial citada y, para usarla, solo requiere la descompresión del archivo descargado en formato zip. Si solo se quisieran ver los detalles más importantes de estos datos, también se puede acceder a ellos de forma sencilla en el sistema operativo Windows, entrando en las propiedades de la imagen, en la pestaña “Detalles”, donde se muestran solo algunos de ellos, pero, en general, los más relevantes.

```

ExifTool Version Number      : 11.08
File Name                    : IMG_180127_103203_0199_RGB.JPG
Directory                   : D:/DocumentosDD/Dany/TFG/Primera_toma
File Size                    : 1140 kB
File Modification Date/Time  : 2018:01:27 10:32:04+01:00
File Access Date/Time       : 2018:07:30 20:22:11+02:00
File Creation Date/Time     : 2018:07:30 12:23:15+02:00
File Permissions             : rw-rw-rw-
File Type                    : JPEG
File Type Extension         : jpg
MIME Type                    : image/jpeg
Exif Byte Order              : Big-endian (Motorola, MM)
Compression                  : JPEG
Photometric Interpretation   : YCbCr
Make                         : Parrot
Camera Model Name            : Sequoia
Orientation                  : Rotate 180
X Resolution                 : 72
Y Resolution                 : 72
Resolution Unit              : inches
Software                     : v1.3.2
Modify Date                  : 2018:01:27 10:32:03
Y Cb Cr Positioning          : Centered
Exposure Time                : 1/1167
F Number                     : 2.3
ISO                          : 100
Exif Version                 : 0210

```

Ilustración 4.1 - Algunos de los datos Exif contenidos en una de las imágenes capturadas para este trabajo. Fuente: Elaboración propia.

4.3. Lenguaje de programación y librería a utilizar

Para realizar los scripts, se requería de un lenguaje de programación que contara con una librería capaz de acceder a estos datos internos de las fotografías y se adecuara a las tareas que se querían realizar. Estudiando las soluciones posibles, en un principio se consideraron las opciones de utilizar el lenguaje de programación Python o C++.

Por un lado, Python es un lenguaje sencillo, fácil de utilizar y muy legible, con una gran cantidad de documentación y librerías que aportan sencillez a la realización

de muchas tareas muy complejas en otros lenguajes. Para acceder con él a los datos Exif, existen varias posibilidades como las librerías “exifread”, “exiftag” o “PIL”.

Por el otro lado, C++ es un lenguaje extremadamente eficiente y también con mucha documentación en línea. Además, se contaba con mayor experiencia personal en este lenguaje comparado con la experiencia en Python. Para acceder a los metadatos Exif de las imágenes también existen posibilidades como “exiv2” y la propia “exiftool”. Sin embargo, usar librerías es una cuestión más compleja en C++ donde, en muchas ocasiones y en este caso, se deben descargar y compilar manualmente, con la dificultad y el coste en tiempo añadidos.

Habiendo considerado ambas opciones, se escogió el lenguaje de programación Python. Se decidió que era la mejor opción porque es un lenguaje más versátil para dos scripts de este tipo, y se creyó que no merecía la pena entrar en la complejidad añadida de utilizar C++ junto con, además, librerías externas. Asimismo, aunque menos que en C++, se contaba con bastante experiencia personal en Python proveniente de trabajos anteriores. A parte, también influyeron en esta decisión el hecho de que la eficiencia no sería un factor crítico, junto a la sencillez, legibilidad y los otros factores mencionados dos párrafos atrás. Por último, Python sería el lenguaje que habrá que usar en la fase de reconstrucción, como se comentará en el capítulo 5.2.1, por lo que sería una buena primera toma de contacto con él en este trabajo.

Así, una vez decidido que el lenguaje a utilizar sería Python, se requería escoger la librería capaz de acceder a los datos Exif para cada imagen. Debido a que no existen demasiadas diferencias entre las distintas posibilidades, a excepción de con PIL que es una librería de imágenes completa innecesaria en este caso, y que únicamente se requerían para acceder a los datos y nada más, se escogió **Exifread**. Esta era la librería que más documentación y soporte tenía mientras se adecuaba perfectamente a nuestras necesidades. Adicionalmente, puede utilizarse junto a las versiones actuales de Python. Para ello, puede instalarse mediante el administrador de paquetes de Python “pip”.

4.4. Selección por altura

En esta selección se buscan eliminar las imágenes tomadas por la cámara multispectral durante los procesos de despegue y aterrizaje del dron. Si se

comprueban los datos Exif de igual forma que en la Ilustración 4.1 y se busca información sobre la altura a la que fue tomada la fotografía, podrá observarse que existe un metadato con nombre “GPS Altitude”, cuyo valor, como se muestra en el ejemplo de la Ilustración 4.2, toma como referencia el nivel del mar. También se podrá comprobar que no hay ningún otro metadato referente a la altura, no existe un metadato con la altura relativa al suelo sobre el que se encontraba el dron.

The image shows a snippet of Exif data. On the left, the text "GPS Altitude" is displayed in a blue, monospaced font. To its right, there is a colon followed by the value "439.4 m Above Sea Level", which is also in a blue, monospaced font.

Ilustración 4.2 - Muestra del dato Exif "GPS Altitude". Fuente: Elaboración propia.

Al no conocer la altura relativa a la que cada imagen fue tomada respecto al punto de despegue, no es posible realizar un filtrado directo que, comprobando cada fotografía, de una sola pasada elimine aquellas realizadas a una altura menor que la altura del vuelo introducida en la aplicación Pix4Dcapture (Ilustración 3.5 e Ilustración 3.6, a la izquierda de la pantalla), que sí toma el punto de despegue como referencia. Sin embargo, podría hacerse con dos pasadas.

Realizando dos pasadas, en la primera habría que comprobar la imagen que fue tomada a la menor altura comparada con las demás. Habría que encontrar el mínimo del valor “GPS Altitude”. De esta forma, este podría tomarse como altura sobre el nivel del mar sobre la cual se realizó el despegue. Una vez conocida, la altura del vuelo sobre el nivel del mar puede conseguirse mediante una simple suma de este valor más la altura relativa del vuelo, que es el que se introdujo en la aplicación Pix4Dcapture. Ya hallada la altura del vuelo sobre el nivel del mar, realizar el filtrado sería una tarea sencilla. Para explicarlo mejor, se supone el siguiente ejemplo:

La primera pasada nos ha permitido ver que la imagen capturada a una menor altura se tomó 200 metros sobre el nivel del mar, por lo que podemos considerar que el punto desde el que el dron partió, o despegó, se encuentra a esta altura. Además, el usuario conoce (o comprueba en Pix4Dcapture, en las propiedades del vuelo realizado) que la altura relativa fijada del vuelo fue de 50 metros. Así, las imágenes de despegue y aterrizaje se encontrarán en el segmento de los 200-250m de altura sobre el nivel del mar, aproximadamente. Sabiendo esto, se realizaría otra pasada sobre las

imágenes y se eliminarían todas aquellas que se encuentren por debajo de los 250 metros, con un pequeño umbral permitido marcado por el usuario.

Algo también a tener en cuenta es que la cámara multispectral, en cada captura, por así decirlo, toma cinco fotografías diferentes al mismo tiempo, una para cada banda espectral (RGB, REG, RED, NIR y GRE). Todas ellas tendrán el mismo valor Exif "GPS Altitude", por lo que, comprobando únicamente las fotografías de una banda específica, no sería necesario comprobar las demás. Como para este tipo de trabajos de fotogrametría, la fotografía en RGB siempre va a estar presente para la reconstrucción, es la que se usa para ello.

De esta forma, el pseudocódigo del script podría ser como el siguiente:

```
altura_despegue = INFINITO

for imagen in fotografias_tomadas:
    if (banda_imagen(imagen) == RGB)
        altura_imagen = tomar_dato_exif("GPS_ALTITUDE")
        if (altura_imagen < altura_despegue):
            altura_despegue = altura_imagen

for imagen in fotografias_tomadas:
    if (banda_imagen(imagen) == RGB)
        altura_imagen = tomar_dato_exif("GPS_ALTITUDE")
        if ( altura_imagen < (altura_despegue + umbral_altura_vuelo) )
            eliminar_imagen_y_bandas(imagen)
```

Ilustración 4.3 - Pseudocódigo del script para el filtrado por altura a la que se tomaron las fotografías.
Fuente: Elaboración propia.

El script final realizado en lenguaje Python se incluye en los anexos. Se detallan a continuación algunos detalles importantes referentes a él:

- Obtiene como argumentos, en primer lugar, la ruta donde se encuentran todas las fotografías que se desean filtrar. Y, en segundo lugar, el valor de altura relativa que marca la altitud mínima que deben tener las imágenes que no se quieran desechar (si la altura del vuelo en Pix4Dcapture era de 50 metros, quizá un buen valor sería 35 metros, permitiendo ese error).
- Para comprobar si la banda en la que se está realizada la imagen es RGB, se aprovecha el hecho de que las fotografías en RGB están tomadas en

el formato JPG, mientras que las demás, en otras bandas, están tomadas en formato TIF, por lo que se buscan únicamente las imágenes en formato JPG.

- Para obtener el dato Exif específico “GPS Altitude” de una imagen utilizando la librería Exifread, así como se obtendría cualquier otro dato, los pasos a seguir son los siguientes:
 - Abrir la imagen en formato binario.
 - Utilizar la función “exifread.process_file(archivo_abierto)”, que devuelve un diccionario *etiqueta, valor*.
 - Buscando en este diccionario la etiqueta “GPS Altitude”, relacionada a él se encuentra el valor de altura deseado.
- Cuando una imagen no cumple el requisito y debe ser eliminada, para eliminar las otras relativas a las distintas bandas espectrales, se tiene en cuenta que cada fotografía está nombrada de la forma “nombreImagen_banda.formato”. Entonces, se eliminan aquellas imágenes que tengan “nombreImagen” = “nombreImagenAEliminar”.

4.4.1. Resultados

La primera toma de imágenes para este trabajo se efectuó con una altura marcada de vuelo de 40 metros, realizándose 2700 fotografías con la cámara multiespectral, contando las cuatro bandas y las RGB. Ejecutando el script sobre este conjunto de imágenes, pasando como argumento primeramente la ruta a la carpeta y después un valor de altura relativa de 25 metros como mínimo, el resultado fue el siguiente:

```

D:\DocumentosDD\Dany\TFG\Scripts\xif>python heightfilter.py D:\DocumentosDD\Dany\TFG\pruebas\Multiespectral 25
-----
Altitud sobre el nivel del mar de despegue encontrada:
436.7684879559791, IMG_180127_104359_0676_RGB.JPG
-----
Imágenes desechadas:
IMG_180127_103033_0139_RGB.JPG --> Altitud 437.512483105472
IMG_180127_103034_0140_RGB.JPG --> Altitud 437.94280678305665
IMG_180127_103036_0141_RGB.JPG --> Altitud 438.53824787166934
IMG_180127_103038_0142_RGB.JPG --> Altitud 439.22144353706545
IMG_180127_103039_0143_RGB.JPG --> Altitud 439.467634778434
IMG_180127_103040_0144_RGB.JPG --> Altitud 439.4517170363683
IMG_180127_103042_0145_RGB.JPG --> Altitud 439.7363419104586
IMG_180127_103043_0146_RGB.JPG --> Altitud 439.79911154783025
IMG_180127_103045_0147_RGB.JPG --> Altitud 439.86406140897265
IMG_180127_103046_0148_RGB.JPG --> Altitud 440.0593868507
IMG_180127_103048_0149_RGB.JPG --> Altitud 440.3393821422413
IMG_180127_103050_0150_RGB.JPG --> Altitud 440.44676450162785
IMG_180127_103051_0151_RGB.JPG --> Altitud 440.27745112370525
IMG_180127_103052_0152_RGB.JPG --> Altitud 439.9223614597626
IMG_180127_103054_0153_RGB.JPG --> Altitud 439.54805048976976
IMG_180127_103055_0154_RGB.JPG --> Altitud 440.33044349968895
IMG_180127_103057_0155_RGB.JPG --> Altitud 441.7591687582246
IMG_180127_103058_0156_RGB.JPG --> Altitud 441.6932038375667
IMG_180127_103100_0157_RGB.JPG --> Altitud 441.93107901149915
IMG_180127_103102_0158_RGB.JPG --> Altitud 442.05125013457194
IMG_180127_103103_0159_RGB.JPG --> Altitud 443.0923079208589
IMG_180127_103104_0160_RGB.JPG --> Altitud 443.12174829213694
IMG_180127_103106_0161_RGB.JPG --> Altitud 443.61953473613886
IMG_180127_103107_0162_RGB.JPG --> Altitud 447.4627153932097
IMG_180127_103109_0163_RGB.JPG --> Altitud 452.14770151413256
IMG_180127_103110_0164_RGB.JPG --> Altitud 456.49774959244377
IMG_180127_103112_0165_RGB.JPG --> Altitud 460.53109064243023
IMG_180127_104013_0526_RGB.JPG --> Altitud 461.1299685656204
IMG_180127_104015_0527_RGB.JPG --> Altitud 460.18874139403
IMG_180127_104016_0528_RGB.JPG --> Altitud 459.3573455167879
IMG_180127_104018_0529_RGB.JPG --> Altitud 459.1983195886407
IMG_180127_104019_0530_RGB.JPG --> Altitud 459.71597513939827
IMG_180127_104021_0531_RGB.JPG --> Altitud 460.6746359516441
IMG_180127_104339_0663_RGB.JPG --> Altitud 460.8741474568138
IMG_180127_104341_0664_RGB.JPG --> Altitud 457.60268639860215
IMG_180127_104342_0665_RGB.JPG --> Altitud 455.4007434062878
IMG_180127_104343_0666_RGB.JPG --> Altitud 453.56937995801576
IMG_180127_104345_0667_RGB.JPG --> Altitud 451.66850392160615
IMG_180127_104347_0668_RGB.JPG --> Altitud 449.51817919783576
IMG_180127_104348_0669_RGB.JPG --> Altitud 447.60774443372867
IMG_180127_104350_0670_RGB.JPG --> Altitud 445.58964501952516
IMG_180127_104351_0671_RGB.JPG --> Altitud 443.63419783612284
IMG_180127_104353_0672_RGB.JPG --> Altitud 441.99948029933114
IMG_180127_104354_0673_RGB.JPG --> Altitud 440.59641040960287
IMG_180127_104355_0674_RGB.JPG --> Altitud 439.1829084457715
IMG_180127_104357_0675_RGB.JPG --> Altitud 438.1325855405557
IMG_180127_104359_0676_RGB.JPG --> Altitud 436.7684879559791
47 Imágenes (235 contando las demás bandas) eliminadas

D:\DocumentosDD\Dany\TFG\Scripts\xif>

```

Ilustración 4.4 - Resultados obtenidos aplicando el filtrado por altura al primer conjunto de imágenes, con una altura mínima indicada de 25 metros. Fuente: Elaboración propia.

Prácticamente todas las fotografías referentes al despegue (las 26 primeras capturas desechadas de la Ilustración 4.4) y aterrizaje (las 13 últimas) fueron eliminadas, junto a otras 6 que durante el vuelo tampoco fueron captadas a la altura suficiente y es correcto que deseché. Todas ellas unidas a sus distintas bandas espectrales, quedando así como resultado 235 imágenes eliminadas en total.

Respecto al segundo conjunto de imágenes que se disponía, éste no fue específicamente captado para este proyecto. La cámara multiespectral capturó imágenes únicamente en las cuatro bandas espectrales NIR, RED, REG y GRE, sin realizar capturas en RGB, por lo que la reconstrucción tendría que hacerse con las

imágenes capturadas por la cámara del dron que, aunque de algo menos de calidad, no dan ningún tipo de problemas. Como se explicó, el dron no realiza capturas hasta que no comienza su trayectoria sobre la superficie marcada, debido a lo cual no fue necesario aplicar esta selección de imágenes de despegue y aterrizaje sobre este conjunto de imágenes.

4.5. Clasificación de imágenes por vuelo

Al realizar varios vuelos distintos, ni la cámara multiespectral ni la cámara del dron las dividen, y finalmente quedan todas agrupadas en las mismas carpetas; todas las imágenes del dron quedan agrupadas en la carpeta de imágenes donde el dron las almacena, y lo mismo ocurre con la cámara multiespectral. Lo que se pretende hacer con esta clasificación es dividir en carpetas distintas las imágenes correspondientes a cada vuelo individual, para que así el usuario no deba dividir las de forma manual, ahorrando el tiempo necesario de clasificar posiblemente varios cientos de imágenes y evitando que pueda cometer más que probables errores confundiendo las imágenes aéreas de una parcela de un terreno con las de otro distinto.

No existe un dato Exif que indique si las imágenes fueron tomadas en un vuelo o en otro distinto. El único factor de clasificación que se puede usar en este caso es la fecha y la hora a la que fueron tomadas las fotografías, que sí podemos encontrar dentro de los metadatos, como se ve en la Ilustración 4.5.



|Date/Time Original : 2018:01:27 10:32:27.481172

Ilustración 4.5 - Muestra del dato Exif "Date/Time Original". Fuente: Elaboración propia.

Gracias a que contamos con esta fecha y hora a la que cada fotografía individual se tomó, podemos aplicar el criterio de que una fotografía pertenece a un vuelo distinto de la inmediatamente anterior fotografía capturada si esta última fue tomada un cierto intervalo de tiempo después. Como las fotografías son capturadas en intervalos de unos segundos, si dos fotografías contiguas fueron tomadas con algunos minutos de diferencia, se puede estar seguro de que son vuelos distintos. Sin embargo, no se debe considerar que esta diferencia de tiempo necesaria para considerar que es un vuelo distinto debe ser muy grande. En el supuesto caso que se quieran realizar dos vuelos inmediatamente seguidos, únicamente cambiando las baterías del dron, es posible que se capture la última fotografía del primer vuelo en el instante justo del

aterrizaje, se cambien las baterías y, en apenas unos minutos de diferencia, cuando el dron esté preparado para el despegue, se comiencen a tomar fotografías de nuevo. Estos vuelos deberían ser considerados distintos, pero si el intervalo de tiempo que se ha indicado para considerar que dos imágenes pertenecen a vuelos distintos, supongamos 15 minutos, es mayor al tiempo que se ha tardado en cambiar las baterías y preparar el nuevo vuelo, supongamos 10 minutos, el script no considerará que hayan sido dos vuelos diferentes y no dividirá las imágenes en carpetas diferentes. Teniendo esto en cuenta, se consideró que una diferencia de dos minutos entre dos imágenes contiguas era buena en el sentido de ser suficiente tanto para considerar que pertenecen a vuelos distintos, como para salvar posibles casos especiales como el mencionado.

Seguidamente, se incluye y explica el pseudocódigo del script buscado:

```
imagenes_RGB = tomar_imagenes(directorio, RGB)

if (imagenes_RGB.estaVacio())
    return

imagenes_ordenadas = imagenes_RGB.ordenarPorFecha()

ultima_fecha = imagenes_ordenadas[0].fecha
#Inicializamos una lista donde guardaremos los índices de la lista que
#apuntan a la última imagen de cada vuelo.
indices_primerasImagenesVuelos = []

for (indice, archivo) in enumerate(imagenes_ordenadas):
    fecha = archivo.fecha
    variacion = ultima_fecha + minutos(2)

    if (fecha > variacion)
        #Variación detectada, vuelos distintos.
        indices_primerasImagenesVuelos.añadir(indice)

    ultima_fecha = fecha

#Para poder ordenarlos después con el método utilizado, se necesita
#añadir el índice de la última imagen.
indices_primerasImagenesVuelos.añadir(imagenes_ordenadas.tamaño)

ultimo_indice = 0
vuelo_actual = 0
for (k, indice_primeraImagen) in enumerate(indices_primerasImagenesVuelos):
    nuevo_directorio = DirectorioActual + '/vuelo_' + vuelo_actual
    crearDirectorio(nuevo_directorio)
    imagenes_vueloActual = imagenes_ordenadas.range(ultimo_indice, indice_primeraImagen)

    if (imagenes_Dron):
        mover_archivos(imagenes_vueloActual, nuevo_directorio)
    else: #Imágenes de la cámara multiespectral
        mover_imagenes_y_bandas(imagenes_vueloActual, nuevo_directorio)

    ultimo_indice = indice_primeraImagen
    vuelo_actual += 1
```

Ilustración 4.6 - Pseudocódigo del script para la clasificación de las imágenes por vuelo. Fuente: Elaboración propia.

Los pasos que se siguen son los siguientes:

1. Recibe como argumentos, primeramente, la ruta dentro de la cual se encuentran las fotografías a clasificar. Además, recibe un parámetro opcional “-b” que, si se indica, significa que este directorio contiene imágenes en distintas bandas espectrales. Si no, se supone que son fotografías individuales captadas por el dron y, en el caso de que también hubiera en otras bandas, no contaría con ellas.
2. Inicialmente, se toman todas las imágenes RGB del directorio.
3. Se ordenan por fecha y hora que fueron tomadas, de menor a mayor.
4. Se inicializa una lista vacía *índices_primerasImágenesVuelos*, donde se guardarán los índices de la lista de imágenes ordenadas donde se encuentren las primeras imágenes de cada vuelo. Además, se crea una variable *ultima_fecha*, que guardará la fecha de la última imagen comprobada, inicializándose a la fecha de la primera imagen de la lista.
5. Para cada elemento de la lista ordenada:
 - a. Se toma su fecha y hora de captura.
 - b. Se comprueba si fue tomada más de dos minutos después de la anterior. Si es el caso, se guarda su índice de la lista de imágenes ordenadas dentro de la lista *índices_primerasImágenesVuelos*.
 - c. Se actualiza el valor *ultima_fecha* al valor de la fecha de la imagen actual comprobada.
6. Se añade a *índices_primerasImágenesVuelos* el índice al último elemento de la lista de imágenes ordenadas.
7. Se crean dos nuevas variables *último_indice* y *vuelo_actual*, inicializadas a cero.
8. Para cada elemento de la lista *índices_primerasImágenesVuelos*:
 - a. Se crea un nuevo directorio o carpeta correspondiente al vuelo actual.
 - b. Se toman todas las imágenes de la lista entre *último_indice* e *índice_primeralimagen*, correspondientes a todas las imágenes del vuelo actual.
 - c. Si las imágenes son de la cámara del dron, se mueven todas estas imágenes directamente. Si, por el contrario, las imágenes son de la

cámara multiespectral, se mueven tanto estas imágenes como las otras que contienen las demás bandas espectrales.

9. Se actualiza $\text{último_índice} = \text{índice_primeraImagen}$ y se suma una unidad a vuelo_actual .

En este caso, con únicamente una pasada sobre todas las fotografías es suficiente. El script final también puede encontrarse en los anexos.

4.5.1. Resultados

El primer conjunto de imágenes utilizado cuenta con, como ya se mencionó, unas 2700 fotografías. La gran mayoría de las imágenes del dron pertenecen a un mismo vuelo, realizado en un terreno en obras cercano a la Universidad de Jaén, donde actualmente se está construyendo el futuro centro comercial “Jaén Plaza”. Sin embargo, también cuenta con otras fotografías tomadas en otros vuelos de pruebas que nada tienen que ver con este terreno y que no deben utilizarse en la fase de reconstrucción. Por otro lado, la cámara multiespectral sí que únicamente se utilizó en el terreno indicado, por lo que el script debe interpretar que todas ellas pertenecen a un único vuelo.

Primero, se aplicó el script a las imágenes de la cámara multiespectral (nótese que se pasó el parámetro “-b”), obteniendo el siguiente resultado:

```
D:\DocumentosDD\Dany\TFG\Scripts\exif>python datefilter.py D:\DocumentosDD\Dany\TFG\pruebas\Multiespectral -b
Directorio: D:\DocumentosDD\Dany\TFG\pruebas\Multiespectral
Vuelos detectados: 1
```

Ilustración 4.7 - Resultado de ejecución del script de clasificación aplicado sobre el primer conjunto de imágenes multiespectrales. Fuente: Elaboración propia.

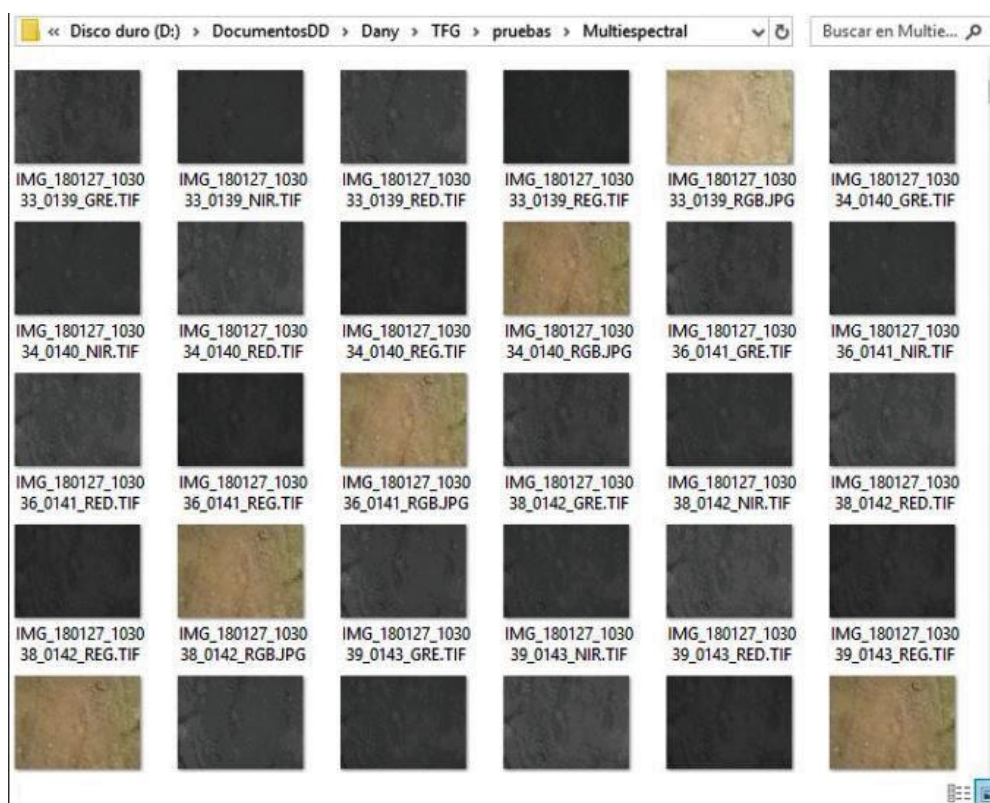


Ilustración 4.8 - Estado de la carpeta antes de aplicar el script de clasificación, como se muestra en la Ilustración 4.7. Fuente: Elaboración propia.

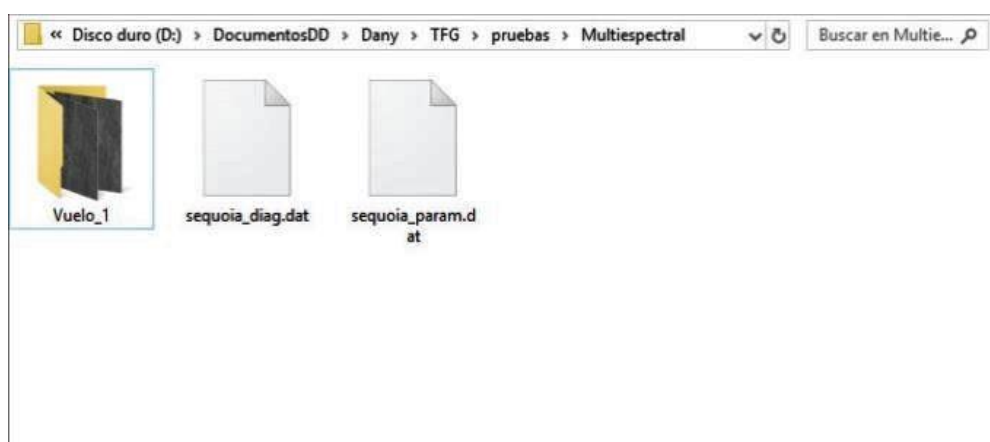


Ilustración 4.9 - Estado de la carpeta después de aplicar el script de clasificación, como se muestra en la Ilustración 4.7. Fuente: Elaboración propia.

Como se puede apreciar, ha detectado de forma correcta un único vuelo, ha incluido todas las imágenes en una misma carpeta “Vuelo_1” y ha dejado fuera dos archivos sueltos creados también por la cámara y que no corresponden a fotografías.

Por último, se aplicó el mismo script a las imágenes del dron donde, como se comentó, sí había fotografías de varios vuelos distintos:

```
D:\DocumentosDD\Dany\TFG\Scripts\exif>python datefilter.py D:\DocumentosDD\Dany\TFG\pruebas\Dron
Directorio: D:\DocumentosDD\Dany\TFG\pruebas\Dron
Variacion detectada D:\DocumentosDD\Dany\TFG\pruebas\Dron\DJI_0241.JPG
Variacion detectada D:\DocumentosDD\Dany\TFG\pruebas\Dron\DJI_0396.JPG
Vuelos detectados: 3
```

Ilustración 4.10 - Resultado de ejecución del script de clasificación aplicado sobre el primer conjunto de imágenes del dron. Fuente: Elaboración propia.



Ilustración 4.11 - Estado de la carpeta antes de aplicar el script de clasificación, como se muestra en la Ilustración 4.10. Fuente: Elaboración propia.

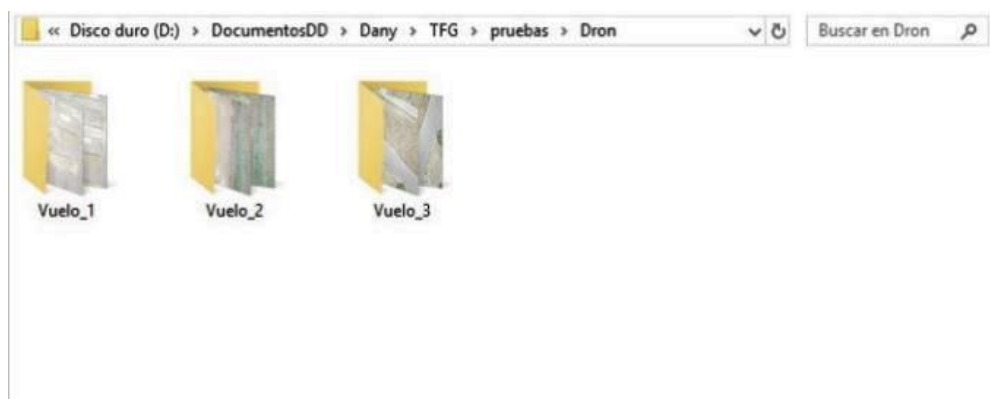


Ilustración 4.12 - Estado de la carpeta después de aplicar el script de clasificación, como se muestra en la Ilustración 4.10. Fuente: Elaboración propia.

Como muestra la Ilustración 4.12, el resultado obtenido es satisfactorio. Ha detectado fotografías de tres vuelos distintos y las ha incluido en sus carpetas correspondientes. Si se entra dentro de ellas, se podrá comprobar que realmente son vuelos distintos.

En los anexos, junto a los scripts, se pueden encontrar también tanto los conjuntos de imágenes iniciales como los resultantes de estas selecciones y clasificaciones para que, si se desea, pueda comprobarse todo lo expuesto en este apartado y, además, los scripts puedan ejecutarse de la misma forma que se ha hecho.

Capítulo 5.

Reconstrucción del terreno.

5. RECONSTRUCCIÓN DEL TERRENO

Una vez terminado todo el proceso de filtrado de imágenes y teniendo ya todas las fotografías clasificadas, es momento de utilizarlas para su verdadero propósito, en las tareas de fotogrametría. En este capítulo, inicialmente, se hablará de los algoritmos “Structure from Motion”, base de las técnicas de fotogrametría utilizadas para la reconstrucción 3D en este tipo de trabajos. Más tarde, se mostrará y analizará la herramienta utilizada para la reconstrucción, PhotoScan. Se explicarán todas las decisiones tomadas en relación a su utilización, así como todos los pasos seguidos para convertir las fotografías en nubes de puntos, modelos 3D, modelos de elevación de terreno, etc. A medida que se avance, también se mostrarán estos resultados obtenidos.

5.1. Técnicas ‘Structure from Motion’ (SfM)

Como se apuntó en el apartado 1.2.4: “Uso de drones para estudios en los campos de la fotogrametría y geomática”, la técnica ‘Structure from Motion’ es un método de bajo coste y efectivo que se basa en principios similares a los de la visión estéreo para reconstruir una estructura tridimensional usando únicamente secuencias de imágenes, resolviendo en este caso el problema de reconstrucción de terreno en el área de la fotogrametría. Antes de indagar específicamente en ella, seguidamente se desarrolla una introducción a la estereoscopia y sus bases. De esta forma, más tarde, se podrá comprender mejor la teoría fundamental de esta técnica.

5.1.1. Visión estereoscópica y cámaras estéreo

A pesar de que el ser humano, al igual que la mayoría de los vertebrados, tiene la sensación de ver solo una imagen, es en realidad su sistema visual el que consigue dar tridimensionalidad a los objetos. Lo consigue a partir de imágenes de cada una de las retinas de los ojos. Es a lo que se llama **estereoscopia** o visión tridimensional. El hecho de tener dos ojos, además de aumentar el campo visual, permite percibir de forma completa las tres dimensiones del espacio que nos rodea.

Para conseguir la visión tridimensional, el cerebro es capaz de apoyarse en ciertas características o elementos integrantes de imágenes estáticas o en movimiento para asociar propiedades de tamaños y posiciones relativas, mediante las cuales, usando las reglas de perspectiva, logra conseguir la apariencia tridimensional. La

visión tridimensional como tal, no requiere que los dos ojos funcionen a la vez, la inteligencia espacial del ser humano siempre asimila las tres dimensiones aun teniendo una mínima capacidad de visión. Sin embargo, la visión tridimensional “completa” o **estereopsis** es la que se obtiene con la ayuda de los dos ojos trabajando al mismo tiempo, gracias a la pequeña disparidad entre las imágenes de cada ojo al enfocar un objeto [35].

En el campo de la Visión por Computador, las imágenes también son bidimensionales mientras pueden estar representando escenas tridimensionales. Durante los años, se ha dedicado una gran cantidad de esfuerzos a cómo conseguir esta tercera dimensión perdida de la mejor forma y, así, obtener la información de profundidad de los elementos que integran la imagen. Hoy día ya existen buenas aproximaciones, pero se le sigue dedicando un alto porcentaje de tiempo [35].

Muestra de ello, las **cámaras estéreo** son uno de los sistemas más básicos para la obtención de profundidad, empleando principios muy parecidos a los de la visión humana. Consisten en dos cámaras alineadas y desplazadas una pequeña distancia una de la otra para que, al capturar las imágenes, se obtengan fotografías también alineadas la una con respecto a la otra.



Ilustración 5.1 - Ejemplo de configuración de cámaras estéreo. Fuente: [36]

Conociendo la distancia a la que se encuentran ambas cámaras y otras propiedades como sus longitudes focales, mediante Geometría Epipolar, en la que se usan las relaciones geométricas entre los puntos 3D individuales de una fotografía y sus correspondientes proyecciones en la otra, y usando técnicas de relación de triángulos, se consigue medir la diferencia de posición, o desplazamiento, de cada entidad tridimensional de una imagen a otra. Si el desplazamiento de un punto de una

imagen a otra es alto, significa que ese punto se encuentra a una distancia cercana de la cámara. Por el contrario, si el desplazamiento ha sido pequeño, ese punto se encuentra alejado de la cámara (Ilustración 5.2). Así, el resultado final se da en forma de un mapa de disparidad, consistente en una representación de las diferentes profundidades a las que se encuentran los objetos con relación a la cámara [35].

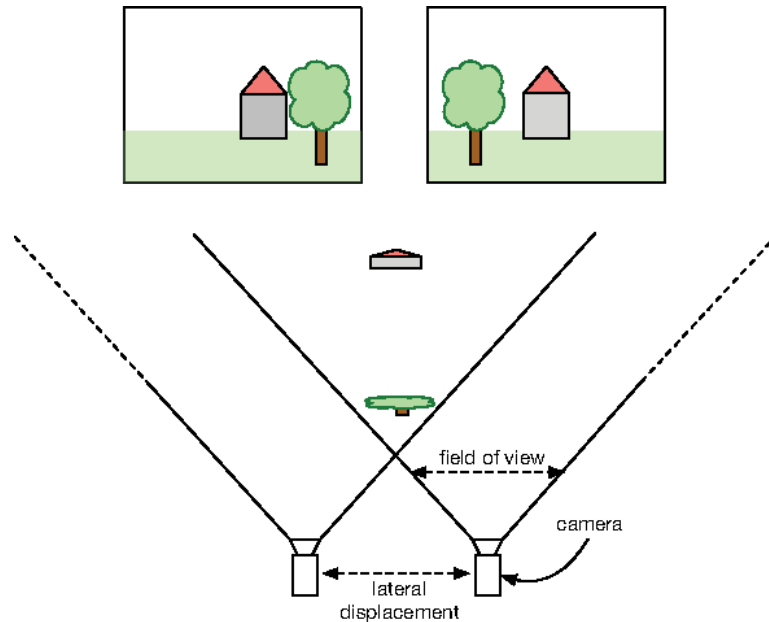


Ilustración 5.2 - Ejemplo de imágenes estéreo. Como se puede observar, el árbol se encuentra mucho más cerca de las cámaras que la casa. Por ello, el desplazamiento del árbol de una fotografía a la otra, es mayor que el desplazamiento de la casa. Fuente: [37]

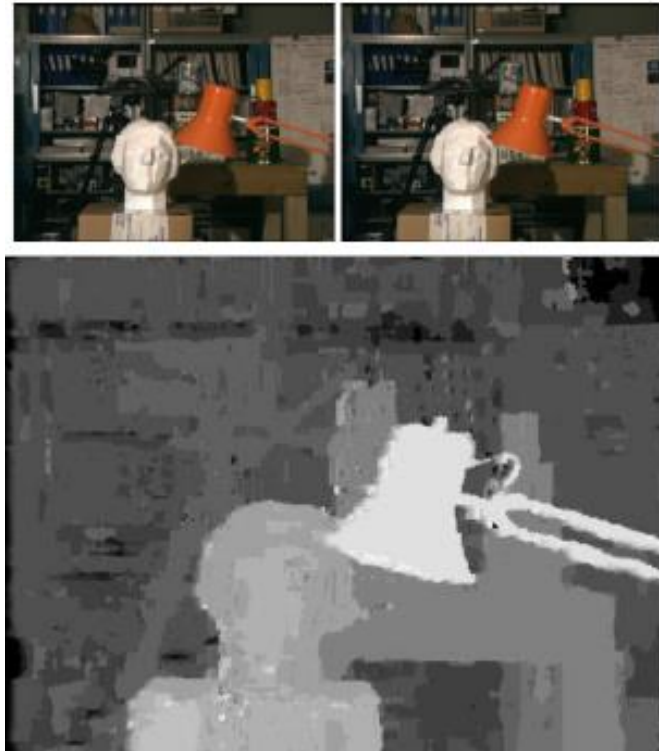


Ilustración 5.3 - Ejemplo de imágenes estéreo (arriba) y mapa de disparidad resultante (abajo). Fuente: [35]

5.1.2. Algoritmos SfM

Los algoritmos SfM usan principios parecidos a los indicados en el apartado anterior. Requieren de un conjunto de imágenes con un alto grado de solapamiento que capturen la estructura tridimensional de la escena desde una amplia cantidad de posiciones. Sin embargo, difieren fundamentalmente de otras técnicas tradicionales en que tanto la geometría de la escena, como también las posiciones de cámara y orientación, se resuelven sin necesidad de conocer ninguna característica de la cámara utilizada. Tampoco es necesario utilizar puntos de control (GCPs), todo se resuelve simultáneamente mediante la identificación automática de características coincidentes en múltiples imágenes. Estas características son buscadas imagen a imagen, permitiendo las primeras estimaciones de las posiciones de la cámara y coordenadas de objeto.

Cabe mencionar que, al no contar con datos de entrada como puntos de control o posiciones GPS, todo se recrea en un sistema de coordenadas relativo, a partir de las imágenes. Si se quisiera transformar a un espacio de coordenadas de mundo, normalmente se realizaría mediante una transformación basada en un pequeño

conjunto de puntos de control de los que sí se conoce sus coordenadas absolutas [13].

Al igual que con la visión estereoscópica, la mayor parte del procesamiento y del peso de estos algoritmos recae sobre la correspondencia, la identificación en distintas imágenes de un mismo punto o misma entidad tridimensional. Dentro de esta fase, el primer paso que se sigue es la identificación de los puntos o características en imágenes individuales que podrían usarse para dicha correspondencia. Para hacerlo, suelen usarse algoritmos como SIFT (Scale Invariant Feature Transform), publicado y patentado por David Lowe [38]. Este algoritmo identifica características en cada imagen que son invariantes a su escala y rotación, y parcialmente invariantes a cambios de iluminación y posición 3D de la cámara. Además, estas son fácilmente distinguibles unas de otras, lo que provee una alta probabilidad de que sus correspondencias sean las correctas entre una gran cantidad de posibilidades. Para conseguirlo, utiliza técnicas basadas en *Diferencias Gaussianas*, consistentes en aplicar un filtro gaussiano con escala σ , para luego aplicar otro filtro del mismo tipo con una escala menor y realizar la diferencia entre ambas, quedándose con los puntos extremos (máximos o mínimos), que son los más resistentes a variaciones [39].

Una imagen de Diferencia Gaussiana $D(x, y, \sigma)$ se define de la siguiente manera [38]:

$$D(x, y, \sigma) = L(x, y, k\sigma) - L(x, y, \sigma),$$

$$L(x, y, \sigma) = I(x, y) * G(x, y, \sigma)$$

Donde:

$I(x, y)$ corresponde a la imagen

$G(x, y, \sigma)$ al filtro Gaussiano, con una desviación estándar σ .

(Fórmula 5.1)

La operación se repite múltiples veces, reduciendo resolución y tamaño de las imágenes para finalmente localizar los puntos realmente invariantes a la traslación, escalado y rotación de la imagen, y mínimamente afectados por el ruido, aplicándoles a cada uno un descriptor basado en su entorno [38]. A parte de SIFT, existen otras

aproximaciones como SURF (Speeded Up Robust Features) [40] u ORB (Oriented FAST and Rotated BRIEF) [41].

A la hora de realizar la reconstrucción 3D, en primer lugar, se buscan y encuentran las correspondencias de cada característica en las otras imágenes, formando “caminos” o “vías” entre los dos puntos. Estas vías deben estar formadas por, como mínimo, dos puntos clave y tres imágenes (Ilustración 5.4). Para encontrar las correspondencias se suelen usar algoritmos del tipo *approximate nearest neighbour*, que radican en encontrar el punto en un conjunto dado que sea más similar a otro punto dado, expresando la cercanía en términos de una función de semejanza [42]; en este caso, usando los descriptores. Cuando ya se tienen estas uniones entre los puntos clave específicos, a través de triangulación y también Geometría Epipolar, se consiguen estimar sus posiciones 3D, creando la llamada nube de puntos dispersa, utilizada para, mediante otro tipo de algoritmos de reconstrucción generalmente basados en mapas de profundidad, ya obtener nube de puntos densa y modelo 3D.

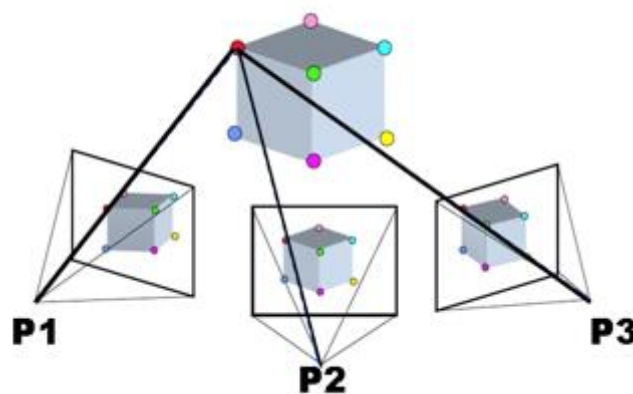


Ilustración 5.4 - Ejemplo del resultado de la correspondencia entre imágenes del algoritmo SfM. Fuente: [43]

5.2. Agisoft PhotoScan

Agisoft PhotoScan es el software de fotogrametría finalmente escogido para realizar este trabajo, capaz de realizar un procesamiento sobre imágenes digitales para generar datos en tres dimensiones. Cuenta con una interfaz sencilla e intuitiva y es usado en aplicaciones de patrimonio cultural, sistemas de información geográfica (SIG), producción de efectos visuales y otras. [44]

Existen dos versiones distintas de este software; la versión profesional y la versión estándar. Ambas son de pago, costando la licencia común para usar la versión profesional 3499\$ y la requerida para utilizar la versión estándar 179\$, aunque también dispone de licencias para uso educativo de un significativo menor precio.

Las principales características de una y otra versión se muestran y comparan en la siguiente tabla:

Característica	Edición profesional	Edición estándar
Triangulación fotogramétrica	Sí	Sí
Generación y edición de nube de puntos densa	Sí	Sí
Generación y texturado de modelo 3D	Sí	Sí
Visión panorámica esférica de 360°	Sí	Sí
Exportación de nubes de puntos y modelo 3D	Sí	Sí
Soporte para cámaras con objetivo de ojo de pez	Sí	Sí
Corrección lumínica de texturas	Sí	No
Clasificación de nube de puntos densa	Sí	No
Exportado de DSM/DTM	Sí	No
Soporte para puntos control	Sí	No
Procesado de imágenes multispectrales	Sí	No
Cálculos de índices de vegetación	Sí	No
Python scripting	Sí	No
Procesamiento en red	Sí	No

Tabla 5.1 - Comparación de características entre las dos versiones disponibles de PhotoScan.
<http://www.agisoft.com/features/compare/>

Como se puede apreciar, este programa congenia perfectamente con los resultados básicos que se buscaban obtener en el comienzo de este trabajo. Únicamente con la edición estándar, mediante técnicas de fotogrametría, ya permite obtener tanto nubes de puntos como modelos en tres dimensiones de la superficie captada con su correspondiente textura. Además, permite su visualización desde todos los ángulos posibles con un total control de la cámara, así como el exportado en distintos formatos.

Sin embargo, uno de los objetivos iniciales de este proyecto era realizar una pequeña fase de “análisis de la superficie”, donde se aprovecharían y utilizarían las imágenes de la cámara multispectral para demostrar solo algunas de las posibilidades que ofrecen. El procesado de imágenes de distintas bandas espectrales no está disponible en la versión estándar de PhotoScan. Sí que lo está en su versión profesional. Además, como también muestra la tabla, permite el análisis en cuanto a cálculos de índices de vegetación.

Igualmente, citando de nuevo los objetivos del proyecto, se decía: “cada fase se intentará automatizar lo máximo posible para simplificar las tareas a un posible usuario que necesite realizarlas”. A diferencia de la versión estándar, la versión profesional permite la ejecución de órdenes internas y la realización de scripts completos mediante el lenguaje Python, a través de una API. En el siguiente apartado, Python API, se desarrolla más sobre ella, pero el poder realizar scripts en lenguaje Python que contengan todas las instrucciones, pasos y órdenes necesarias es ideal para cumplir este objetivo. Permite la automatización de todos los pasos generales que deben dar la mayoría de usuarios para este tipo de trabajos.

Por último, dentro de las características de la versión profesional, cabe destacar también la capacidad de procesamiento en red. Esta podría permitir la ejecución remota de todas las tareas -muy costosas en tiempo y en recursos- en una máquina o clúster remoto, ayudándose también de la API. Incluso podría permitir el procesamiento en la nube. Precisamente, en algunas partes de este trabajo se hará la ejecución de las tareas en una máquina remota, ya que fue permitida la utilización del clúster Turing de la Universidad de Jaén para ello.

Por todo lo indicado, PhotoScan fue el programa elegido para este proyecto. Específicamente, se escogió su versión profesional (Agisoft Photoscan Professional Edition), que cuenta con todas las características y ventajas mencionadas.

5.2.1. Python API

Como se ha comentado, PhotoScan, en la versión profesional utilizada, dispone de una API en lenguaje Python. Esta se encuentra en desarrollo, con la última versión, a julio de 2018, lanzada el día 5 de este mismo mes (*Release 1.4.3*).

Hay que decir que, en general, es algo confusa y complicada de utilizar. Además, no existe demasiada documentación disponible sobre ella, a parte de la oficial y un foro dentro de la propia página web de Agisoft. Sin embargo, es mucha la funcionalidad a la que se puede acceder [45]:

- Abrir, crear y guardar proyectos de PhotoScan.
- Añadir *chunks*, fotografías y puntos de control. PhotoScan permite, por así decirlo, crear varias líneas de trabajo distintas en cada proyecto. “Chunk” es el nombre que recibe cada uno de estos flujos de trabajo individual. Se pueden crear, copiar, mezclar y eliminar.
- Añadir y modificar calibraciones de cámara, datos de control, asignar proyecciones y coordenadas geográficas.
- Ejecutar los distintos pasos del procesamiento (alineación de fotografías, creación de la nube de puntos densa y malla, textura, simplificar el modelo, etc.)
- Exportar los resultados (modelos, texturas, ortofotos, modelos digitales de elevaciones, etc.)
- Acceso a los datos de las imágenes, así como de los modelos y nubes de puntos creados.
- Controlar las tareas de procesamiento en red.

Como se puede notar, todas las operaciones básicas para este tipo de estudios de fotogrametría están disponibles y son accesibles desde el *Python scripting*, además de permitir el control del procesamiento en red.

El módulo principal de esta API tiene como nombre “PhotoScan”. Desde él se puede acceder a todas las clases y, en general, a toda funcionalidad. Su principal componente es la clase “Document”, que representa un proyecto de PhotoScan. A su vez, dentro de un objeto “Document”, se puede acceder a sus distintos chunks, si se tiene más de uno, y aplicarles ya todas las etapas del procesamiento [45].

Otra clase importante es “Application”, que da acceso a varios atributos globales de la aplicación; por ejemplo, el objeto “Document” en uso. Se puede acceder a ella desde el atributo “PhotoScan.app”.

Por último, cabe destacar que, para ejecutar comandos individuales o scripts, se dispone tres opciones: ejecutar comandos desde el propio PhotoScan, que dispone de un panel como consola de comandos interna; ejecutar scripts también desde el propio PhotoScan, que dispone dentro del menú “Herramientas” de la opción “Ejecutar secuencia de comandos”; o desde una línea de comandos usando el argumento “-r” y pasando también la ruta al script como argumento.

Después de estudiar la API, así como realizar diversas pruebas, se determinó la estructura que debe tener un script genérico que realice el flujo de trabajo básico en PhotoScan:

```

1  import glob
2  import PhotoScan
3  import time
4
5  def main():
6      doc = PhotoScan.Document()
7
8      chunk = doc.addChunk()
9      chunk.label = "Main_Chunk"
10
11     path_photos = "Vuelo_1/*.JPG"
12
13     #checking save filename
14     project_path = "PS_projects/vuelo_01_reconstruction.psx"
15
16     #adding photos
17     chunk.addPhotos(glob.glob(path_photos))
18
19     #align photos
20     chunk.matchPhotos(accuracy=PhotoScan.HighAccuracy,
21                     preselection=PhotoScan.GenericPreselection)
22     chunk.alignCameras()
23
24     #build dense cloud
25     chunk.buildDepthMaps(quality=PhotoScan.HighQuality,
26                       filter=PhotoScan.ModerateFiltering)
27     chunk.buildDenseCloud()
28
29     #build model
30     chunk.buildModel(surface=PhotoScan.HeightField,
31                   interpolation=PhotoScan.EnabledInterpolation)
32     #build texture
33     chunk.buildUV(mapping=PhotoScan.GenericMapping)
34     chunk.buildTexture(blending=PhotoScan.MosaicBlending, size=4096)
35
36     #export dense point cloud
37     chunk.exportPoints(project_path[:-4]+'ptcloud.obj',
38                     format=PhotoScan.PointsFormatOBJ)
39
40     doc.save(project_path)
41
42     print("Script finished")
43     return 1
44
45 if __name__ == "__main__":
46     t0 = time.time()
47     main()
48     t1 = time.time()
49     total = t1 - t0
50     print("total time =", total, "seconds")

```

Ilustración 5.5 - Script que ejecuta el flujo básico de trabajo en PhotoScan. Fuente: Elaboración propia.

Seguidamente, se explica su funcionamiento, así como los métodos y funciones utilizados [46] [45]:

- **doc = PhotoScan.Document()**. En primer lugar, se debe crear un objeto *Document* que, como se comentó, representa un proyecto de PhotoScan. A través de la clase *PhotoScan*, se accede a su constructor.
- **chunk = doc.addChunk()**. Es necesario añadir un objeto *Chunk* para este proyecto, derivado del objeto *Document* antes creado. Se debe guardar en una variable.
- **chunk.label = "New Chunk"**. Es recomendable añadir una etiqueta al *chunk*, ya que se podría crear más de uno y sería necesario diferenciarlos.
- **chunk.addPhotos(glob.glob(path_photos))**. Una vez definido el *chunk*, se le añaden todas las fotografías. La función "addPhotos" de la clase *Chunk* se define de la siguiente forma en la documentación oficial (Ilustración 5.6):

```
addPhotos (filenames[, layout], strip_extensions=True[, progress])
```

Add a list of photos to the chunk.

Parameters

- **filenames** (*list of string*) – A list of file paths.
- **layout** (*PhotoScan.ImageLayout*) – Image layout in the chunk.
- **strip_extensions** (*bool*) – Strip file extensions from camera labels.
- **progress** (*Callable[[float], None]*) – Progress callback.

Ilustración 5.6 - Definición de la función "addPhotos" de la clase *Chunk* en la API de PhotoScan.
Fuente: [45].

Como se puede apreciar, el parámetro **filenames** espera una lista de strings conteniendo las rutas a todas las fotografías a utilizar. Crear una lista manual con la ruta a cientos o miles de fotografías es algo inviable, más aún cuando es el propio usuario quien debe especificarla para cada proyecto distinto. Por ello, se decidió usar la biblioteca externa de Python "glob", también utilizada en los scripts de filtrado. Con ella, pasando un patrón de una ruta relativa o absoluta, se consigue una lista de strings que contienen todas las rutas de archivos específicos que cumplen con ese patrón. En el caso de la Ilustración 5.5, se pasó el patrón de ruta relativa "Vuelo_1/*.JPG", obteniendo como resultado una lista de rutas a las fotografías JPG, que coinciden con las RGB, de la carpeta "Vuelo_1".

- **chunk.matchPhotos(.)**. Con ya las fotografías incluidas, este método pone en marcha el algoritmo para hallar las correspondencias entre ellas¹. Recibe, inicialmente, un parámetro indicando la precisión que se desea y que se debe escoger entre las constantes disponibles en *PhotoScan.Accuracy*. A más precisión, mayor calidad, pero mayor tiempo de procesamiento necesario.
También se puede indicar si se desea preselección de pares, y de qué tipo. Esta preselección se utiliza para amenizar el proceso de hallar correspondencias. Mediante la preselección genérica que se le indica en el script de muestra, PhotoScan no busca correspondencias entre todas las imágenes, sino que inicialmente realiza una detección rápida de pares de imágenes que cuentan con un grado notable de superposición entre ellas y únicamente busca correspondencias entre estos pares. En este tipo de trabajos con superficies no supone ninguna desventaja, ya que se cuenta con un grado de solapamiento muy alto y no es necesario buscar correspondencias entre todos los cientos de imágenes.
Además, se pueden añadir otros parámetros opcionales, como el máximo de correspondencias entre imágenes, pero para este tipo de trabajos no son necesarios.
- **chunk.alignCameras()**. Ordena que se realice el cálculo de las posiciones de las cámaras y, con ello, la creación de la nube de puntos dispersa.
- **chunk.buildDepthMaps(.)**. Calcula los mapas de profundidad de la escena. Para crear la nube de puntos densa, PhotoScan debe calcular antes la información de profundidad para cada cámara. Se le debe pasar como parámetro la calidad deseada, utilizando para ello las constantes disponibles en *PhotoScan.Quality*.
Además, se puede indicar el modo de filtrado. El modo de filtrado indica a PhotoScan cómo actuar frente a puntos con posible ruido o borrosidad en las imágenes. Se escoge entre las constantes de *PhotoScan.FilterMode*. En este caso

¹ PhotoScan utiliza algoritmos de correspondencia similares a SIFT, detectando puntos en las fotografías estables a variaciones del punto de observación e iluminación y generando descriptores para cada punto basados en su vecindario, pero algo diferentes para un alineamiento de mayor calidad [44].

se aplica un nivel moderado, ni agresivo ni pasivo, utilizado en la mayoría de los casos.

- **chunk.buildDenseCloud()**. Construye la nube de puntos densa.
- **chunk.buildModel(.)**. Construye el modelo 3D. El parámetro más importante es “surface”, que indica si es una superficie arbitraria *-Arbitrary-* o si es una superficie planar *-HeightField-*, como terrenos o superficies, y que debe ser el utilizado para la fotogrametría aérea, ya que requiere menor cantidad de memoria y permite el procesamiento de mayores conjuntos de datos. Además, se puede indicar si permitir la interpolación o, por el contrario, no y, si no hay información suficiente en una zona de la nube de puntos densa, no trazar el modelo sobre ella.

Hasta aquí el procesamiento básico para trabajos de fotogrametría aérea. De forma opcional, se puede continuar para realizar otros análisis. Se muestra como ejemplo el cálculo de textura del terreno. También, para poner un ejemplo, se realiza un guardado o exportación de la nube de puntos densa, así como se podrían exportar todos los demás estudios generados en distintos formatos posibles. Asimismo, PhotoScan no provee detalles del tiempo de procesamiento total de un trabajo. Únicamente, en la interfaz gráfica, mientras se está realizando una tarea, muestra el tiempo actual y una estimación del tiempo restante. Sin embargo, utilizando la API, mediante la biblioteca externa *Time* de Python, sí se puede calcular el tiempo final, como se detalla.

5.3. Metodología seguida y resultados

PhotoScan es un programa con una interfaz gráfica muy sencilla e intuitiva, con una división de espacios basados en paneles fácil de comprender. Como se ha visto, el flujo básico de trabajo general consiste en añadir las fotografías, orientarlas, con lo que ya se crea la nube de puntos dispersa; y crear la malla. Todo ello en el orden descrito. Después, si se requiere, también es posible crear textura, modelo digital de elevaciones, ortomosaico, etc.

5.3.1. Primera prueba

En primer lugar, con el fin de realizar unos primeros test con este programa y ver los primeros tiempos de procesamiento, se utilizó la interfaz gráfica para introducir

unas imágenes correspondientes al parking del rectorado de la universidad. Estas también fueron captadas por el dron, pero suman únicamente 40 capturas correspondientes al despegue, ascenso y estabilización del dron en el aire, descenso y aterrizaje.

En el ordenador de sobremesa utilizado, que monta un procesador Intel Core i5-750, 8 GB de memoria RAM y una tarjeta gráfica NVIDIA GeForce GT 220, aplicando unos parámetros generales de calidad alta, el tiempo de procesamiento total, contando la orientación de cámaras, nube de puntos densa y malla, fue de aproximadamente 1.5 horas. Solo viendo el resultado de la nube de puntos dispersa, ya se observa el previsto mal resultado que se obtuvo (Ilustración 5.7).

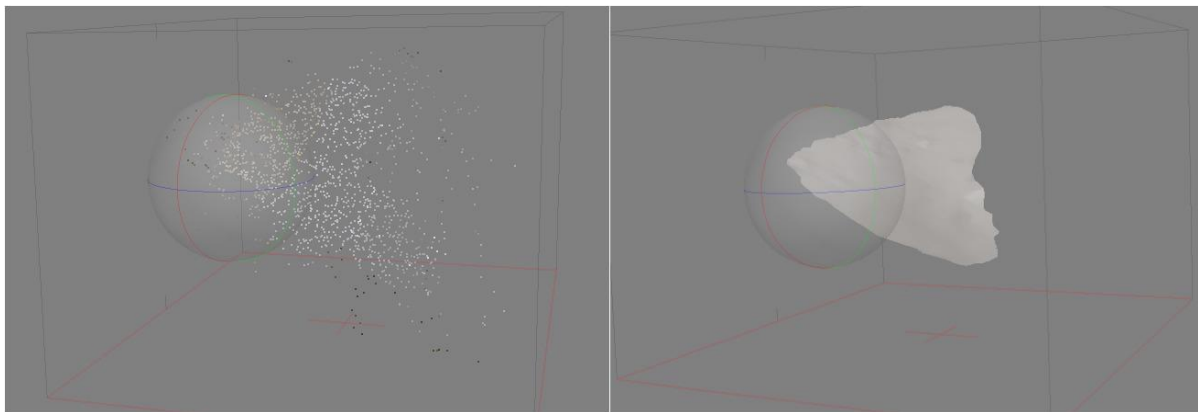


Ilustración 5.7 - Resultado de prueba inicial. A la izquierda, nube de puntos dispersa; a la derecha, modelo 3D.

El parking del rectorado es una zona llana, pero la reconstrucción resultante no tiene esta forma. En primer lugar, tiene un grado de inclinación muy alto. En segundo lugar, en la reconstrucción aparecen muchas elevaciones y hundimientos. Por último, la forma original tampoco coincide con la forma triangular de la reconstrucción.

De forma clara, las fotografías contaban con un alto grado de solapamiento, debido a que prácticamente el dron solo realizó un ascenso y descenso en vertical. Sin embargo, al no captar la escena desde distintas posiciones en el plano horizontal y, además, ser captadas en alturas muy distintas, los algoritmos SfM no fueron capaces de identificar correctamente la escena. No están diseñados para trabajar con imágenes de estas características. Aun así, se obtuvieron los primeros datos en cuanto al coste en tiempo de ejecutar estos procedimientos.

5.3.2. Primera ejecución con el conjunto de imágenes del terreno en obras (calidad media)

El primer conjunto de imágenes sí tomadas para este trabajo, después de la fase de filtrado, quedó con unas 490 fotografías RGB en total, correspondientes al terreno en obras frente a la universidad. El procesamiento de éstas iba a ser mucho más largo que el de la primera prueba en el mismo ordenador. Por ello, para este estudio, ya sí se utilizó la API de PhotoScan, ejecutada en el clúster. Hay que decir que el clúster tenía bastante carga de trabajo en general durante todas las distintas ejecuciones para este trabajo. Se podían observar picos de uso de, a veces, hasta 90%-100% de memoria y CPU.

Sabiendo que la ejecución, a pesar de ser ejecutada en el clúster, seguiría requiriendo un largo tiempo, se decidió dividir el script en tres partes. La primera consistiría en la creación del proyecto, añadido de fotografías y alineación de cámaras, creando la nube de puntos dispersa. En la segunda, se construirían los mapas de profundidad y la nube de puntos densa. Por último, en la tercera, se construiría el modelo y textura. En cada uno de ellos, además, se exportarían los resultados obtenidos. De esta forma, se conseguiría ir obteniendo resultados poco a poco, pudiendo comprobar que la ejecución y el resultado de cada una de las fases principales es correcta, decidiendo así pasar a la siguiente. En el caso de que fuera incorrecta, se podría repetir y se podrían cambiar diversos parámetros como los referentes a la calidad, para así evitar una cadena de malos resultados consecutivos.

También, se decidió que la calidad general que se pasaría como argumento, con las distintas constantes correspondientes a los distintos métodos, fuese media. Ni muy alta o alta, ni baja, sino un punto medio entre todas. De esta forma, se podría ver correctamente el potencial general de estos algoritmos al mismo tiempo que el tiempo sería relativamente corto.

El script de la primera parte ejecutada consiste básicamente en el script mostrado en la Ilustración 5.5, eliminando las líneas 23-39. Es decir, quitando la construcción de la nube de puntos densa, el modelo, la textura y la exportación de la nube de puntos densa; y cambiando la calidad a media. En la Ilustración 5.8 se puede comprobar el resultado de esta ejecución en el clúster:

```
36 weak
adjusting: xxxxxxxxxxxxxxxxxxxxxx 1.18673 -> 1.165
point variance: 1.45361 threshold: 4.36084
adding 18931 points, 29652 far (4.36084 threshold), 51 inaccurate, 2 invisible,
46 weak
adjusting: xxxxxxxxxxxxxxxxxxxxxx 1.03676 -> 1.0159
point variance: 1.24963 threshold: 3.74888
adding 23156 points, 30599 far (3.74888 threshold), 16 inaccurate, 6 invisible,
56 weak
adjusting: xxxxxxxxxxxxxxxxxxxxxx 0.92447 -> 0.905113
point variance: 1.09926 threshold: 3.29778
adding 23370 points, 28708 far (3.29778 threshold), 10 inaccurate, 1 invisible,
135 weak
adjusting: xxxxxxxxxxxxxxxxxxxxxx 0.839568 -> 0.827375
point variance: 0.994412 threshold: 2.98324
adding 22733 points, 26740 far (2.98324 threshold), 18 inaccurate, 2 invisible,
147 weak
optimized in 561.485 seconds
coordinates applied in 0 sec
SaveProject
saved project in 7.36639 sec
Script finished
total time = 64467.87752985954 seconds
[gggi@turing33 ~]$
```

Ilustración 5.8 - Resultado en consola del cálculo de nube de puntos dispersa en el primer conjunto de imágenes. Fuente: Elaboración propia.

El tiempo total de ejecución fue de 64467 segundos, es decir, unas 17.9 horas. En la siguiente ilustración, se puede ver el resultado en forma de la nube de puntos dispersa:

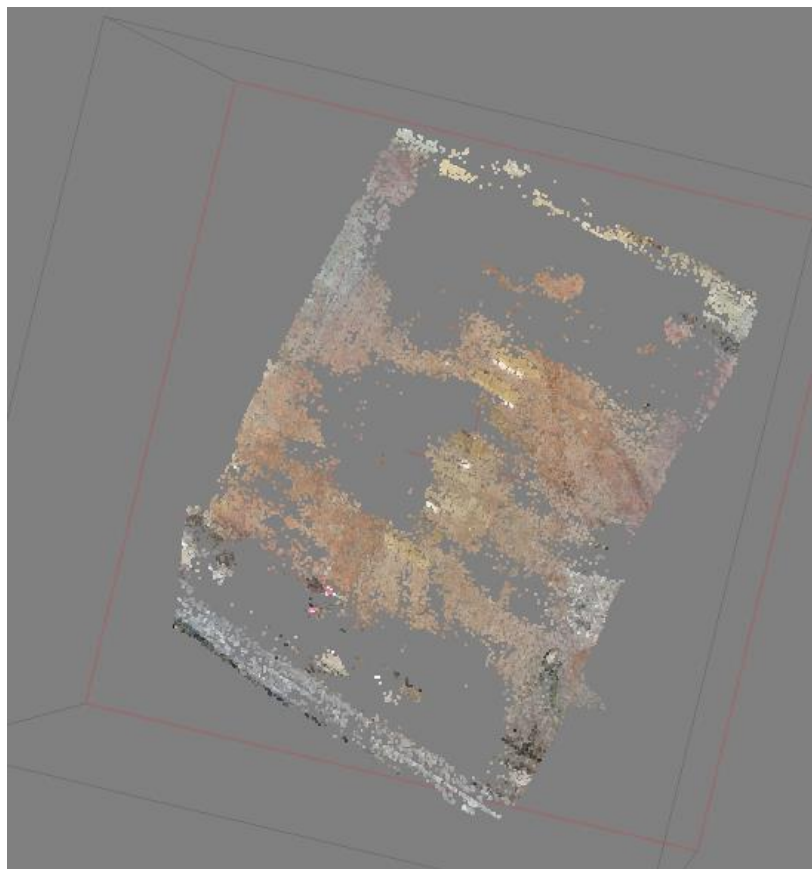


Ilustración 5.9 - Resultado del cálculo de nube de puntos dispersa en el primer conjunto de imágenes. Fuente: Elaboración propia.

Como se observa, el resultado en general parece bueno. Hay unas zonas que han sido reconstruidas mejor, con muchos puntos como referencia, al mismo tiempo que en otras parece no haber a penas. Sin embargo, aunque pudiera parecer que no, en la mayoría de las zonas menos con menos densidad, siguen existiendo puntos que, de seguro, son útiles para los siguientes cálculos. En la siguiente ilustración, se puede ver desde otra perspectiva:

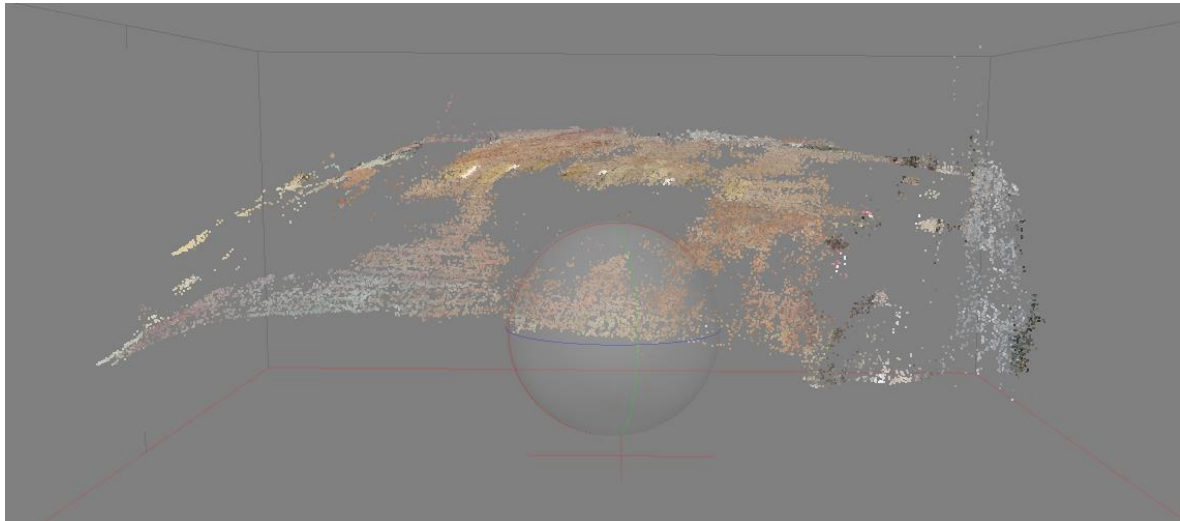


Ilustración 5.10 - Resultado del cálculo de nube de puntos dispersa en el primer conjunto de imágenes, visto de frente. Fuente: Elaboración propia.

El resultado tiene una inclinación que parece que no es correcta, con forma convexa. Sin embargo, se decidió continuar con la ejecución, en lugar de repetirla con otros parámetros ya que, en general, la nube de puntos parece correcta en densidad. También, para poder comprobar el resultado final de la calidad media.

A continuación, el script de la segunda parte consiste en la creación de los mapas de profundidad y de la nube de puntos densa. Se muestra también el resultado de la ejecución en consola:

```
6 fill: 1e-06

Depth reconstruction devices performance:
- 100% done by CPU
Total time: 42367.8 seconds

BuildDenseCloud
selected 409 cameras in 3.61385 sec
working volume: 5918x6026x2641
tiles: 1x1x1
selected 409 cameras in 0.003809 sec
preloading data... done in 7.25028 sec
filtering depth maps... done in 3107.53 sec
preloading data... done in 19.3786 sec
accumulating data... done in 54.3134 sec
building point cloud... done in 8.49899 sec
36744737 points extracted
SaveProject
saved project in 105.048 sec
LoadProject
loaded project in 20.1068 sec
Script finished
total time = 45755.99804878235 seconds
[gggi@turing33 ~]$
```

Ilustración 5.11 - Resultado en consola del cálculo de nube de puntos densa en el primer conjunto de imágenes. Fuente: Elaboración propia.

El tiempo de ejecución de esta parte fue de 45756 segundos, es decir, 12.71 horas. En la siguiente ilustración también se ve el resultado final:

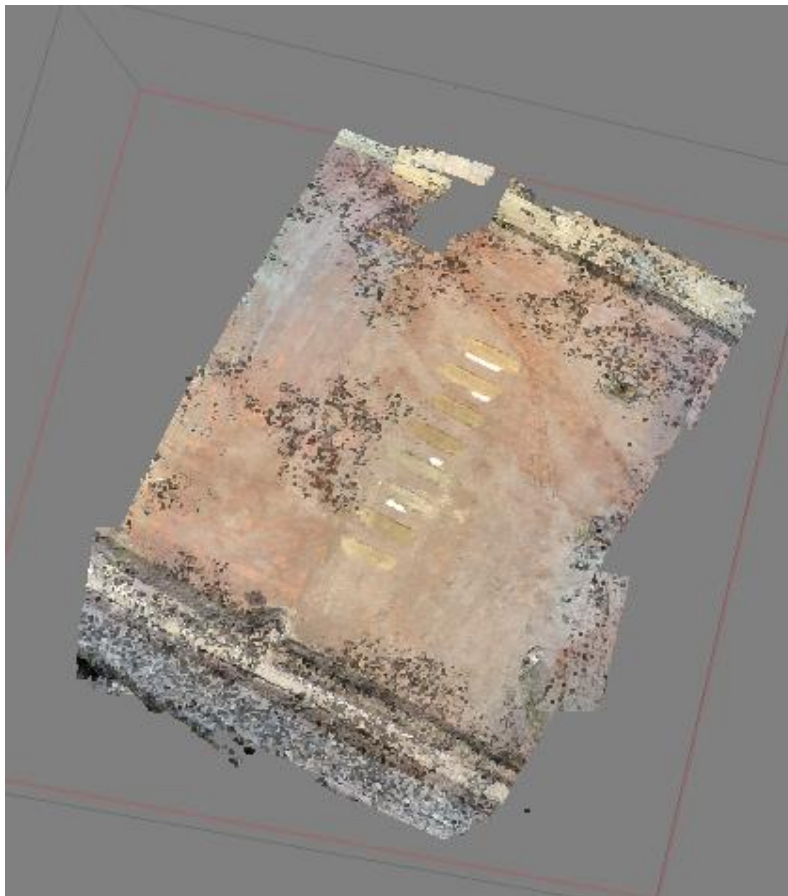


Ilustración 5.12 - Resultado del cálculo de nube de puntos densa en el primer conjunto de imágenes. Fuente: Elaboración propia.

La mayor parte de los huecos se han rellenado correctamente, aunque se ven diversas zonas con errores que se corresponden con aquellas donde en la nube de puntos dispersa no halló tantas correspondencias. Con estos resultados se puede comenzar a comprobar lo que devuelve un procesamiento de calidad media.

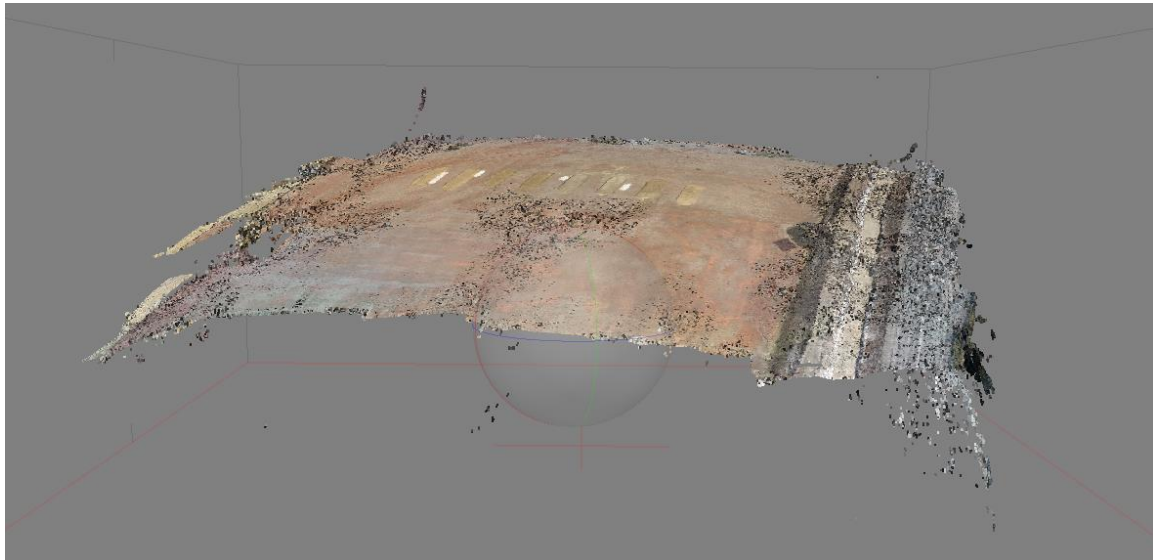


Ilustración 5.13 - Resultado del cálculo de nube de puntos densa en el primer conjunto de imágenes, visto de frente. Fuente: Elaboración propia.

Completada la nube de puntos densa, únicamente faltaba la ejecución del cálculo del modelo o malla junto a su textura. Se muestra, seguidamente, el resultado de esta última ejecución en el clúster.

```
*?***** done in 451.282 sec  
blending textures... ?????????????????????????????*****  
*****  
*****  
*****  
*****  
*****?  
?*????????*??????????  
postprocessing atlas... done in 466.676 sec  
  
ExportPoints  
SaveProject  
saved project in 52.2189 sec  
LoadProject  
loaded project in 15.151 sec  
Script finished  
total time = 174682.7070159912 seconds  
[qgqi@turing33 ~]$
```

Ilustración 5.14 - Resultado en consola del cálculo del modelo 3D y textura en el primer conjunto de imágenes. Fuente: Elaboración propia.

El tiempo de ejecución de esta última parte fue mucho más alto. En concreto, 174683 segundos, es decir, 48.5 horas. En primer lugar, este mayor tiempo se debe

a que la propia naturaleza de estos cálculos los hace más costosos que los anteriores. Pero también es consecuencia de que, durante este proceso, el clúster mostraba aún más carga de otros trabajos y disponía de aún menos recursos, presentando en los momentos anteriores a la ejecución y durante ella, un uso de CPU y memoria de prácticamente el 100%. A pesar de todo, el resultado final fue el siguiente:

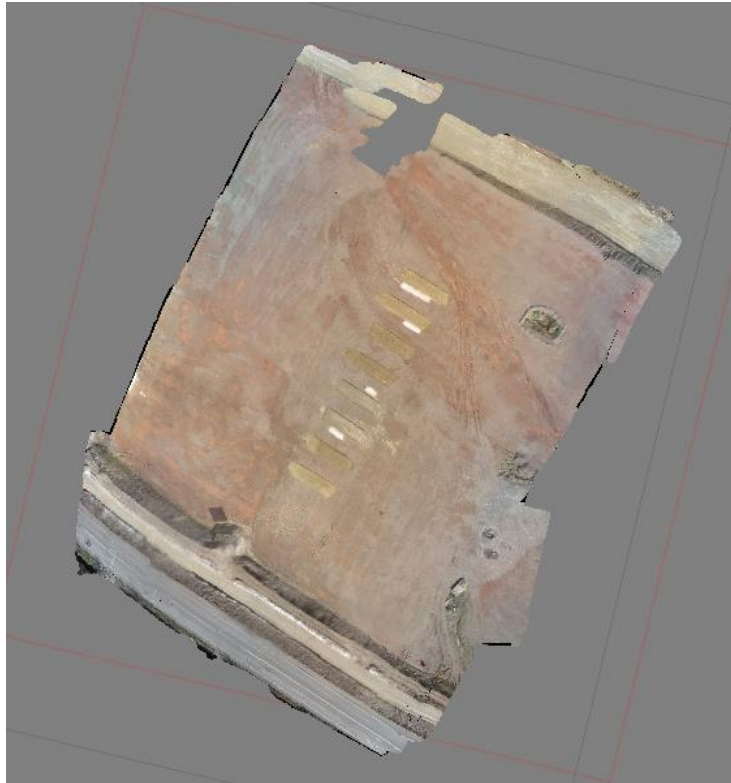


Ilustración 5.15 - Resultado del cálculo de modelo y textura en el primer conjunto de imágenes. Fuente: Elaboración propia.

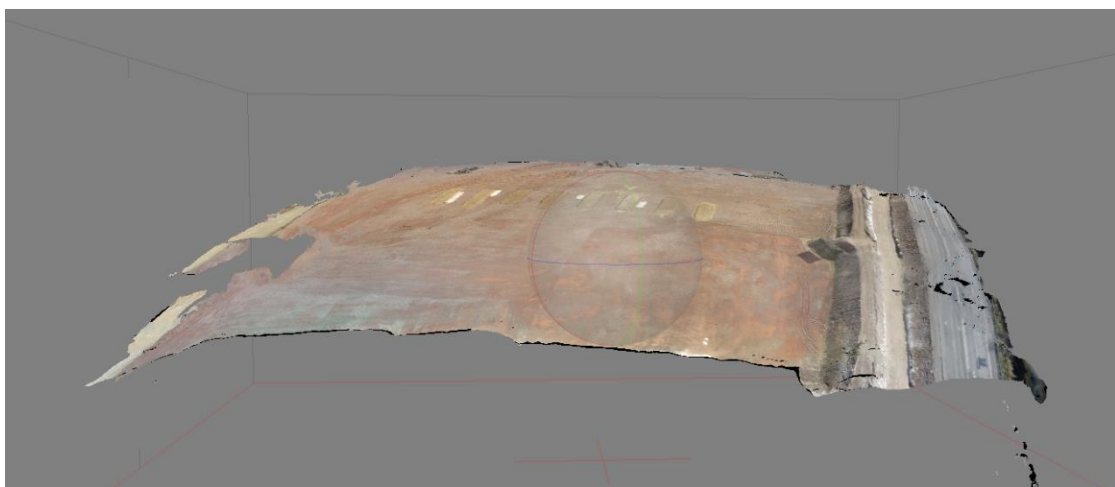


Ilustración 5.16 - Resultado del cálculo de modelo y textura en el primer conjunto de imágenes, visto de frente. Fuente: Elaboración propia.

En este modelo resultante, con su correspondiente textura, se han corregido muchos de los que parecían ser errores o zonas con artefactos. El resultado final es bastante bueno, aunque sigue existiendo esa curvatura que no refleja el terreno llano real.

5.3.3. Segunda ejecución con el conjunto de imágenes del terreno en obras (calidad alta)

Una vez visto el resultado obtenido con calidad media, se procedió a realizar el mismo procesamiento, pero con parámetros de calidad alta -por debajo de muy alta, la calidad máxima-. Para este estudio, debido a la excesiva carga de trabajos que seguía presentando el clúster y habiendo obtenido un tiempo de procesamiento de más de dos días para la última fase de la prueba anterior, se decidió realizar la ejecución directamente en el ordenador de sobremesa, a través de la interfaz gráfica. El tiempo de procesamiento de todo el flujo de trabajo duró unos siete días y los resultados son los siguientes:

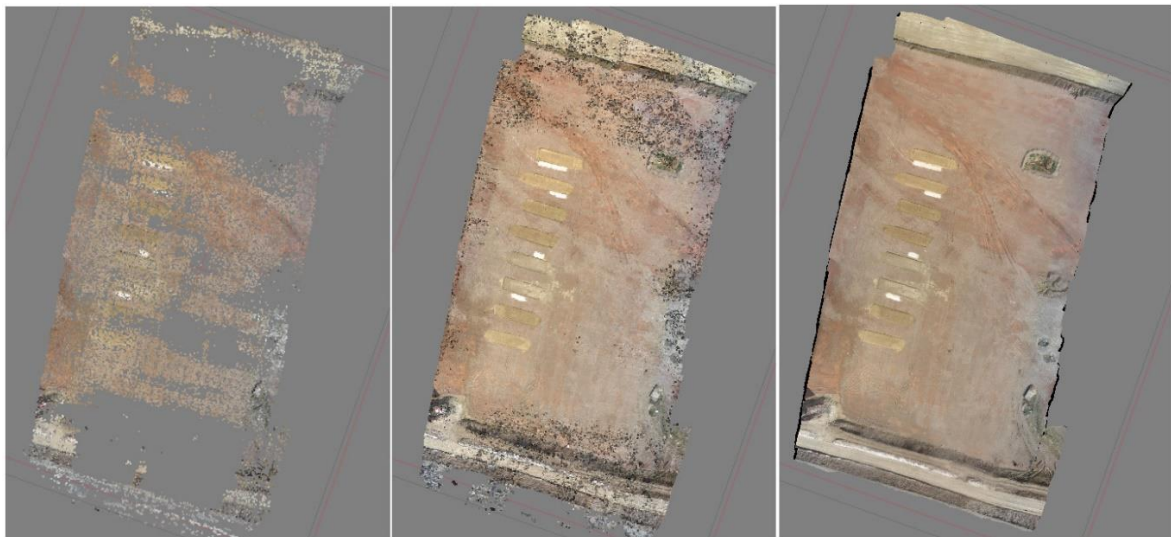


Ilustración 5.17 - Resultados en forma de nube de puntos dispersa (izquierda), nube de puntos densa (centro) y modelo con textura (derecha) de la ejecución con parámetros de calidad alta sobre el conjunto de imágenes del terreno en obras, vistos desde arriba. Fuente: Elaboración propia.

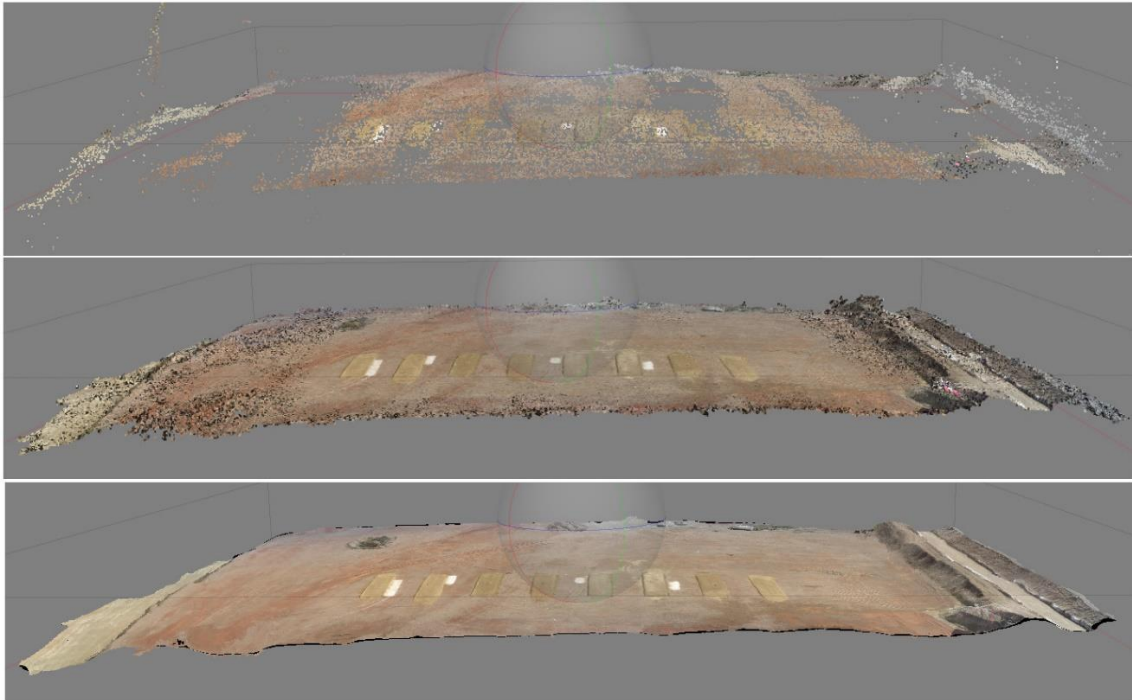


Ilustración 5.18 - Resultados en forma de nube de puntos dispersa (arriba), nube de puntos densa (centro) y modelo con textura (abajo) de la ejecución con parámetros de calidad alta sobre el conjunto de imágenes del terreno en obras, vistos desde frente. Fuente: Elaboración propia.

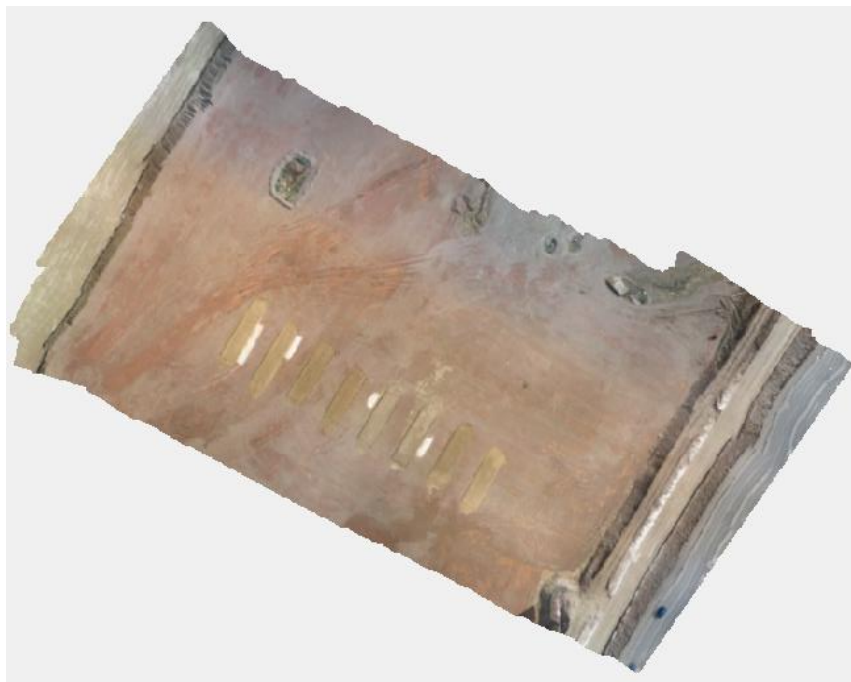


Ilustración 5.19 - Resultado en forma de ortomosaico de la ejecución con parámetros de calidad alta sobre el conjunto de imágenes del terreno en obras. Fuente: Elaboración propia.

Como se aprecia, el resultado es de mucho mayor calidad comparado al anterior. La primera nube de puntos ya cuenta de por sí con una mayor densidad, tapando muchos huecos respecto a su equivalente anterior. Además, se ha eliminado esa concavidad artificial y ahora la reconstrucción sí refleja la naturaleza real de

elevaciones del terreno captado. Se aprecia mucho mejor en el modelo digital de elevaciones también calculado y obtenido:

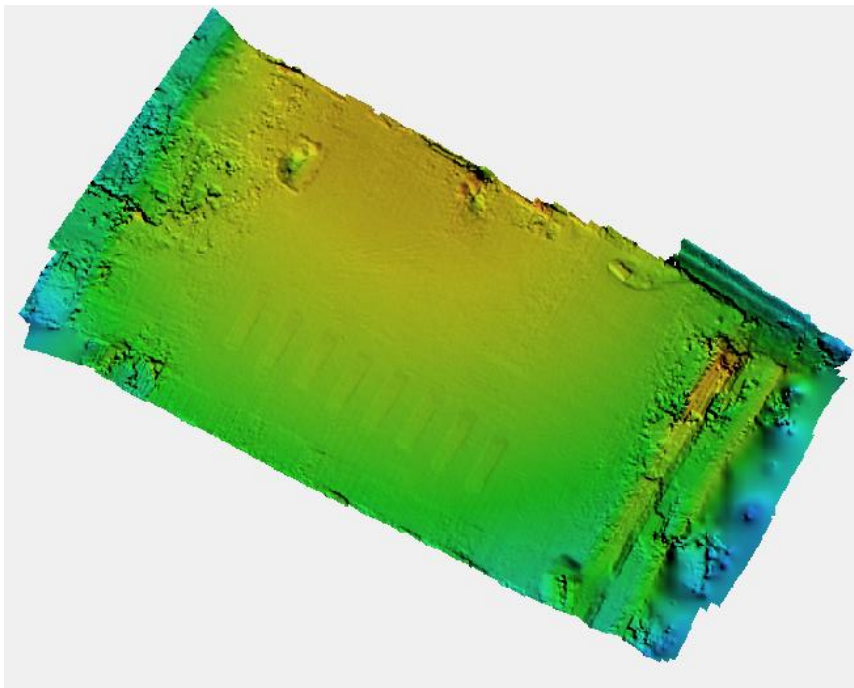


Ilustración 5.20 - Resultado en forma de MDE de la ejecución con parámetros de calidad alta sobre el conjunto de imágenes del terreno en obras. Fuente: Elaboración propia.

A causa del buen resultado ya obtenido y al alto tiempo de procesamiento requerido tanto en este ordenador de sobremesa específico como en el clúster, se decidió no hacer de nuevo la reconstrucción con, esta vez, parámetros de calidad muy alta. En lugar de ello, se resolvió hacer una reconstrucción con calidad muy alta, pero utilizando ya para ello el segundo conjunto de imágenes, bastante más pequeño que el actual. Se describe en el apartado que sigue.

5.3.4. Ejecución con el segundo conjunto de imágenes (calidad muy alta)

El segundo conjunto de imágenes corresponde a una finca. El nombre que recibe la zona es “Torrecillas”, por ello muchos de los archivos resultantes de este estudio contienen ese nombre. Como recordatorio, a este conjunto de imágenes no le fue necesario aplicar el filtrado por altura, pues las imágenes RGB fueron directamente captadas por el dron, lo que hace que, de por sí, no fueran tomadas fotografías durante el despegue y el aterrizaje. El total de fotografías asciende a 130, alrededor de una cuarta parte del total de imágenes del primer conjunto, por lo que realizar una reconstrucción con parámetros de calidad muy alta, la máxima, sería correspondientemente más accesible.

En esos momentos, el clúster, a pesar de seguir teniendo una carga de trabajo considerable, volvía a presentar un uso algo menor. A causa de ello, se decidió ejecutar este proyecto también en él. Esta vez se dividiría el procesamiento en dos partes. Primeramente, se ejecutaría la carga y alineación de fotografías junto al cálculo de la nube de puntos densa. En segundo lugar, el cálculo de modelo y textura.

No se disponen de datos de tiempo de la ejecución de la primera parte, debido a que, después de su finalización, surgió un error en la consola que no permitió capturar este valor. El fallo se dio únicamente en la ventana con la instancia de la consola. La propia ejecución terminó de forma correcta y los resultados no sufrieron ningún error. Por otra parte, sí se cuenta con el dato correspondiente a la segunda parte de la ejecución, el modelo y textura. El tiempo requerido fue de 61764 segundos, es decir, 17.16 horas. Los resultados se exponen seguidamente:

```
-----  
loaded project in 33.5663 sec  
Script finished  
total time = 61763.55128240585 seconds  
[gggi@turing33 ~]$
```

Ilustración 5.21- Resultado en consola del cálculo del modelo 3D y textura en el conjunto de imágenes de Torrecillas con calidad muy alta. Fuente: Elaboración propia.



Ilustración 5.22 - Resultados en forma de nube de puntos dispersa (izquierda), nube de puntos densa (centro) y modelo con textura (derecha) de la ejecución con parámetros de calidad muy alta sobre el conjunto de imágenes de Torrecillas, vistos desde arriba. Fuente: Elaboración propia.

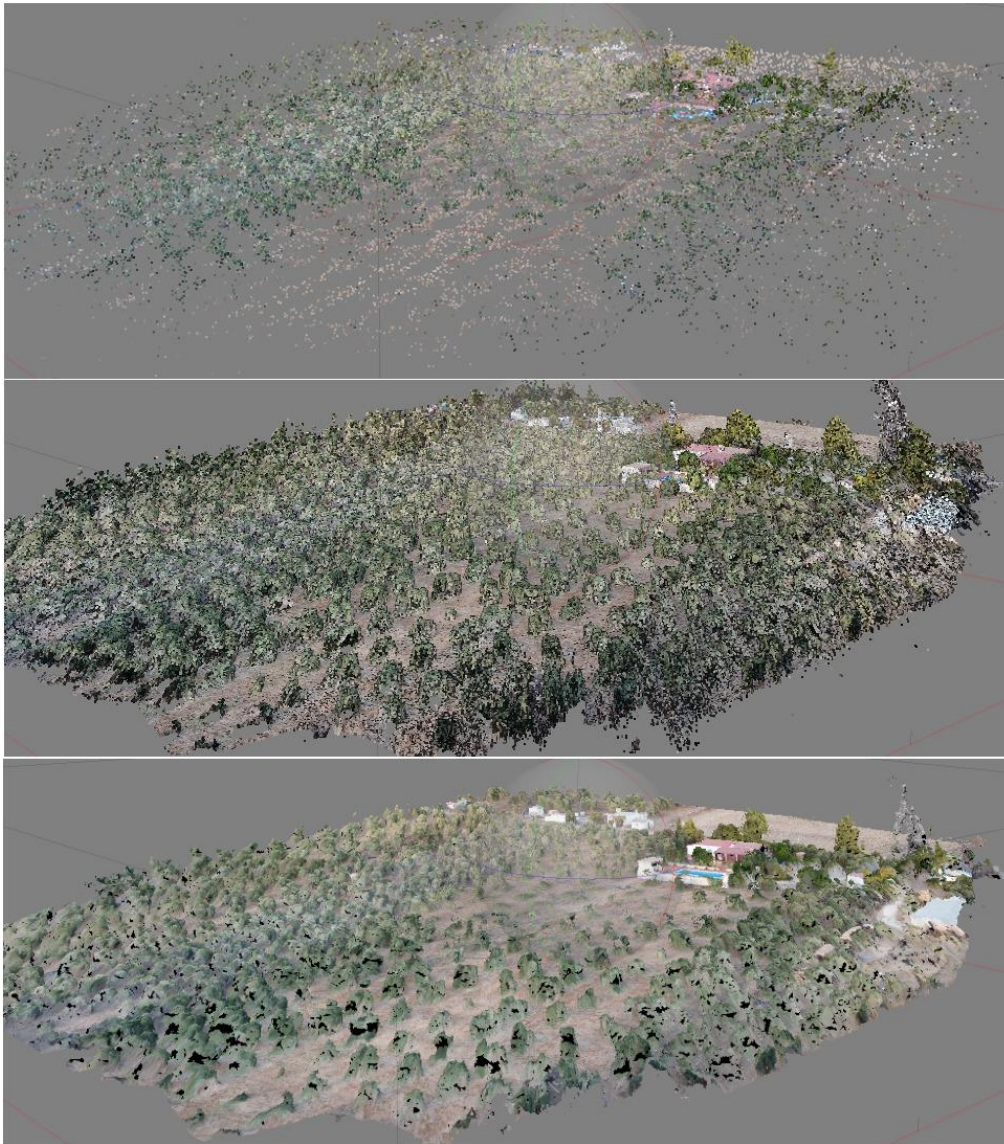


Ilustración 5.23 - Resultados en forma de nube de puntos dispersa (arriba), nube de puntos densa (centro) y modelo con textura (abajo) de la ejecución con parámetros de calidad muy alta sobre el conjunto de imágenes del terreno de Torrecillas, vistos desde un lado. Fuente: Elaboración propia.



Ilustración 5.24 - Detalle del modelo y textura obtenidos como resultado de la ejecución con parámetros de calidad muy alta sobre el conjunto de imágenes del terreno de Torrecillas. Fuente: Elaboración propia.

Los resultados, vistos desde arriba, parecen ser muy buenos y tener una gran calidad. Todas las construcciones, olivos e incluso los coches dan la sensación de haber sido reconstruidos muy bien, lo que se espera de una ejecución con calidad muy alta. No obstante, si se cambia de perspectiva, se observa cómo, en realidad, los olivos no han sido modelados del todo bien, teniendo muchas zonas oscuras en la zona media y baja sin reconstruir (Ilustración 5.23). Esto se debe a que las imágenes han sido captadas desde arriba y, aunque se hayan capturado desde muchos puntos distintos, por su naturaleza no proveen información de los olivos a parte de sus copas, en la mayoría de los casos. Utilizando estas técnicas que únicamente usan las imágenes aéreas, es muy difícil reconstruir correctamente la vegetación de forma íntegra. Sobre todo, aquellos árboles de poca altura y muy frondosos en la copa y zona media, como los olivos, ya que tapan al completo la parte baja y el tronco (Ilustración 5.25). Pese a ello, a la hora de realizar análisis de la vegetación, o del olivar en este caso, a través de imágenes multiespectrales, no es un problema como se verá en el capítulo 6: Índices de vegetación. Tampoco sería un problema si se quisiera realizar un análisis aún mayor.



Ilustración 5.25 - Fotografía de muestra del conjunto de imágenes del terreno de Torrecillas, que ilustra la dificultad del dron para captar la parte baja de los olivos. Fuente: Conjunto de imágenes utilizado.

A parte del problema descrito, la calidad de la reconstrucción es la esperada. Lo que es la superficie en sí, ha sido modelada de forma muy precisa y sin errores destacables. Además, el modelo digital de elevaciones expone la naturaleza del desnivel del terreno de forma correcta (Ilustración 5.26). Sin duda, es la mejor reconstrucción de todas las realizadas y sin requerir un tiempo excesivo comparado con ellas. Si se desea ver con más detalle, en los anexos se adjunta el proyecto de PhotoScan junto al modelo exportado en formato 'obj' y la nube de puntos densa en formatos 'obj' y 'ply'. Hay que decir que son archivos muy pesados, por lo que pueden tardar un cierto tiempo en cargar en programas externos de modelado.

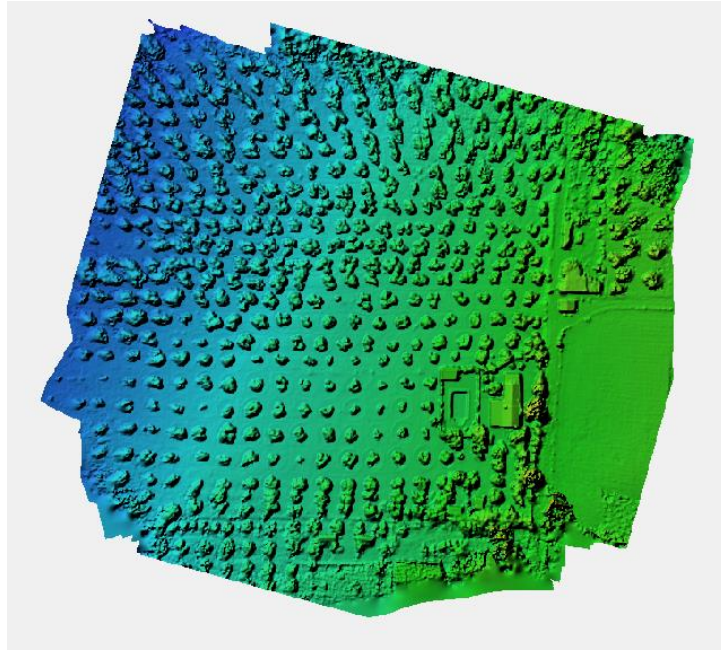


Ilustración 5.26 - Resultado en forma de MDE de la ejecución con parámetros de calidad muy alta sobre el conjunto de imágenes del terreno de Torrecillas. Fuente: Elaboración propia.

Todos los proyectos, conjuntos de imágenes y resultados están disponibles en los anexos de este trabajo. También las imágenes mostradas en este documento, incluidas aquellas individuales que se han usado para hacer las composiciones (ver capítulo 9. A).

Capítulo 6.

Índices de vegetación.

6. ÍNDICES DE VEGETACIÓN

En este capítulo se pretende demostrar, a parte de la ya vista capacidad de reconstrucción de las superficies con muy buena calidad, la también gran capacidad de investigación y análisis que proveen los estudios de fotogrametría aérea con cámaras capaces de captar distintas bandas espectrales. En concreto, se centra en el cálculo de índices de vegetación utilizando estas imágenes.

Los índices se definen, en este caso, como un conjunto de operaciones algebraicas realizadas sobre los valores numéricos de los píxeles de imágenes captadas en dos o más bandas espectrales sobre una misma escena. De forma específica, un índice de vegetación se puede definir como un parámetro calculado a partir de los valores de la reflectancia a distintas longitudes de onda, y que es particularmente sensible a la cubierta vegetal (Gilabert et al, 1997). De forma general, los valores altos de índices de vegetación apuntan a una vegetación fuerte, mientras que los valores bajos indican una vegetación débil [47]. A continuación, se estudian y aplican tres índices de vegetación distintos sobre las imágenes del segundo conjunto, las tomadas sobre la finca de Torrecillas. Cabe mencionar que, en los anexos, también se incluye una pequeña prueba realizada al principio con apenas unas imágenes del primer conjunto.

Debido a que son trabajos muy poco costosos, todo se realizó utilizando directamente la interfaz gráfica de PhotoScan. La metodología seguida ha consistido en, inicialmente, crear un nuevo proyecto de PhotoScan, tomar las imágenes multiespectrales y añadirlas, al igual que si fueran fotografías normales. En ese momento, PhotoScan detecta que son imágenes en distintas bandas, por lo que aparece una ventana para seleccionar cómo se desean importar: creando una cámara a partir de cada archivo, como si fueran fotografías distintas; o creando cámaras multiespectrales a partir de archivos como bandas, creando una cámara para cada fotografía compuesta por las cuatro bandas distintas. Se selecciona la segunda opción y se sigue el flujo de trabajo normal hasta conseguir el ortomosaico. En ese momento ya se pueden realizar y visualizar los cálculos de índices. Se accede a esta utilidad de cálculos de índices desde el menú, en “Herramientas -> Set Raster Transform”, donde se permite introducir cualquier fórmula para el cálculo del índice deseado (Ilustración 6.1). El resultado se refleja en el ortomosaico.

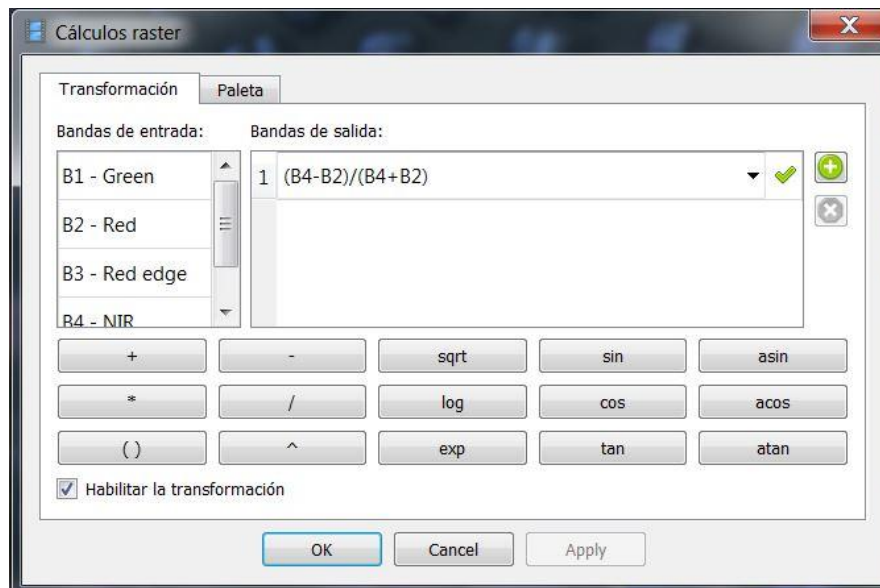


Ilustración 6.1 - Ventana que se muestra en PhotoScan al seleccionar en el menú "Herramientas" y, seguidamente, "Set Raster Transform". Permite realizar cálculos con las distintas bandas espectrales.
Fuente: Elaboración propia.

6.1. RVI

El índice RVI (Ratio Vegetation Index) es el pionero de entre todos los índices de vegetación. Fue propuesto en 1972 por Pearson y Miller y se calcula como el cociente entre la reflectividad del infrarrojo cercano y la del rojo. Es poco sensible a las condiciones de iluminación, pero mucho a las propiedades ópticas de la tierra. En este caso, el valor resultante es inversamente proporcional a la cantidad de vegetación presente en el área. La fórmula, como se ha comentado, es la siguiente:

$$RVI = NIR / RED.$$

(Fórmula 6.1)

6.1.1. Resultados

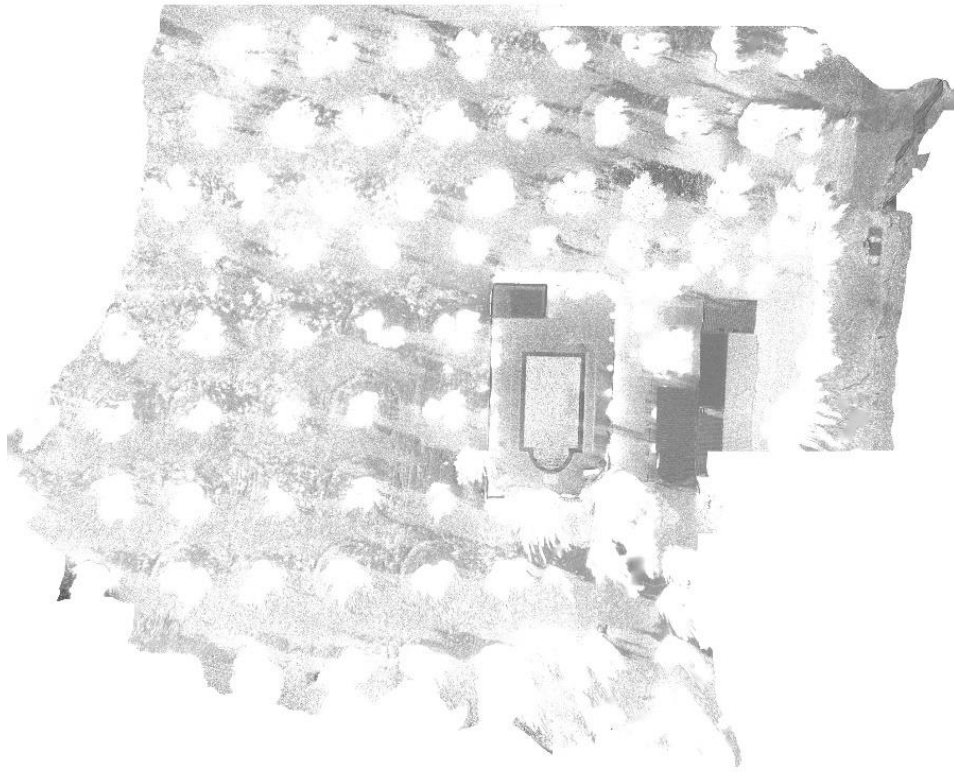


Ilustración 6.2 - Resultado del cálculo RVI en el proyecto de imágenes multiespectrales. Fuente: Elaboración propia.

Al ser el primero de todos los índices de vegetación definidos, se observa que el resultado no está demasiado refinado. Habiendo usado una escala de grises para visualizar los resultados, es cierto que, los olivos, aparecen claramente más claros que la tierra. Pero también, por ejemplo, la reflectancia del agua de la piscina ha provocado que esa zona también resulte más clara. Además, debido al resultado tan claro o luminoso general de la imagen, no se consigue diferenciar de forma definida unas zonas de otras. Con una escala de representación distinta, o aplicando algún tipo de filtro a la imagen, se podría conseguir una mejor visualización. En todo caso, los resultados obtenidos con este índice no se encuentran siempre en un mismo rango de valores, por lo que cada problema podría ser distinto. A pesar de todo, especialmente si se ve la imagen a pantalla completa, el resultado general es bueno.

6.2. NDVI

NDVI (Normalized Difference Vegetation Index) es el índice de vegetación más extendido en la actualidad. En general, mucho más que todos los demás. Se utiliza en

todo tipo de aplicaciones debido a su sencillez de cálculo y el resultado final en un rango fijo entre -1 y 1, lo que permite establecer umbrales y comparar imágenes. De forma aproximada, un valor de 0 representaría el valor aproximado donde empieza a existir vegetación y, los positivos, representan superficies con vegetación. Se calcula de la siguiente forma:

$$\text{NDVI} = (\text{NIR} - \text{RED}) / (\text{NIR} + \text{RED}).$$

(Fórmula 6.2)

6.2.1. Resultados



Ilustración 6.3 - Resultado del cálculo NDVI en el proyecto de imágenes multiespectrales. Fuente: Elaboración propia.

El resultado general es mucho más claro que el anterior. PhotoScan provee de forma directa la escala normal para representar los índices NDVI y otros derivados. Esta toma el valor 0 como color blanco, simulando ausencia de vegetación; 0.1 marrón, imitando el color de la tierra con vegetación muy pobre; y entre 0.3 y 0.6 distintos valores de verde de más claro a más oscuro. Utilizándola, el resultado es mucho más vistoso y no es de extrañar que sea el índice de vegetación más usado

hoy en día. Toda la vegetación está perfectamente representada en color verde y, de un solo vistazo, se entiende su distribución a lo largo y ancho de la escena.

6.3. GNDVI

GNDVI (Green Normalized Difference Vegetation Index) es un índice derivado del anterior y muy parecido a él. La única diferencia es que, en este caso, no se usa la banda del rojo, sino la del verde, lo que la hace especialmente útil para la determinación de la captación de agua y nitrógeno de las plantas.

$$\text{GNDVI} = (\text{NIR} - \text{GREEN}) / (\text{NIR} + \text{GREEN}).$$

(Fórmula 6.3)

6.3.1. Resultados



Ilustración 6.4 - Resultado del cálculo GNDVI en el proyecto de imágenes multiespectrales. Fuente: Elaboración propia.

Los resultados son muy parecidos al anterior. Además, se dan en el mismo rango de valores y, por lo tanto, se utiliza la misma escala de representación. Si se comparan fijamente, se observa que se dan pequeñas variaciones entre este índice y NDVI, pero, en este caso, muy poco significativas. Ambos producen resultados muy buenos.

Capítulo 7.

Prototipo de aplicación para visualizar los resultados.

7. PROTOTIPO DE APLICACIÓN PARA VISUALIZAR LOS RESULTADOS

En este capítulo, se explica el objetivo final que se propuso para este TFG: la realización de un prototipo software para la visualización de los resultados obtenidos. Inicialmente, se justifica la toma de esta decisión. Seguidamente, se hace un análisis de requisitos y de los resultados que se pretenden obtener. Por último, se comenta la parte del desarrollo de este prototipo y el resultado final.

7.1. Justificación

Una vez realizado todo el proceso de reconstrucción y estudio, los resultados obtenidos eran muy variados y heterogéneos. En muchas ocasiones, se encontraban desperdigados entre varias carpetas. Por ello, estudiando la situación, se vio como interesante la idea de una aplicación que unificara todo el trabajo realizado. Un programa que el usuario pudiera utilizar para visualizar y navegar fácilmente sobre las fotografías finales utilizadas durante el proceso de reconstrucción, así como los resultados de estos procedimientos y del análisis.

Además, se conocía que PhotoScan permite exportar en fichero de texto las posiciones virtuales desde las que el dron realizó cada captura de las fotografías, así como otros parámetros relativos a la cámara en ese instante. Estos datos también podrían usarse en la aplicación final para permitir al usuario ver las posiciones y orientación del dron en cada captura. Y, gracias a que los datos se encuentran ordenados desde la primera fotografía tomada hasta la última, podría incluso realizarse un *travelling* o recorrido virtual de la trayectoria que siguió el dron durante el vuelo. De esta forma, se le podría ayudar a darse cuenta de ciertos errores que podría haber cometido en la fase de captura de imágenes o de algunas mejoras que podría aplicar. Por ejemplo, si viera que las posiciones de las capturas están relativamente distantes unas de otras, quizá sería una buena idea cambiar algunos valores durante la planificación del recorrido como la velocidad del dron o el periodo entre dos capturas distintas de fotografías, para que el recorrido seguido sea así más eficiente y permita un mayor nivel de solapamiento entre las fotografías.

Así, se marcó como objetivo final para este TFG el realizar un prototipo de esta aplicación, con la que se pudieran visualizar, para cada proyecto, todas las fotografías iniciales y todos los resultados obtenidos, así como detalles sobre el vuelo realizado.

Seguidamente se pasa a hablar sobre todo lo relacionado con ello. En primer lugar, se realizará un análisis de los requisitos que se buscan que tenga este prototipo, tanto funcionales como no funcionales; después se reflexionará sobre la plataforma a escoger para realizarlo y, más tarde, se entrará en detalle sobre el desarrollo y el resultado final.

7.2. Análisis de requisitos

A continuación, se enumeran los requisitos que debería cumplir la aplicación final y que deben ser representados en el prototipo a realizar.

7.2.1. Requisitos funcionales

La aplicación debe ser capaz de visualizar, para cada proyecto:

- El conjunto de imágenes inicial, tanto las fotografías RGB como las multiespectrales.
- Los resultados que se hayan obtenido:
 - Modelos 3D, mallas o nubes de puntos.
 - Resultados en forma de imágenes.
- Para los resultados en forma de modelos 3D, mallas o nubes de puntos, debe permitir trasladar y rotar la cámara virtual a través de la escena según las acciones indicadas por el usuario mediante el ratón como dispositivo de entrada.

Debe poder mostrar o ejecutar, a petición del usuario, los siguientes detalles relacionados con el vuelo del dron:

- Orientación y posición del dron desde las que se realizaron las capturas de las fotografías utilizadas.
- Recorrido virtual de la trayectoria seguida por el dron durante el vuelo en el que se realizaron las capturas de las fotografías utilizadas.

7.2.2. Requisitos no funcionales

Los requisitos no funcionales son aquellos que se refieren a la calidad, usabilidad, rendimiento, etcétera del producto. En este caso, se seleccionan los siguientes:

- Debido a que se trabaja con archivos de modelos 3D y nubes de puntos bastante grandes, los tiempos de carga no deben ser excesivos.
- El usuario debe poder mover, rotar y hacer *zoom* con la cámara de forma natural, sin que esta realice saltos, movimientos bruscos o no naturales, en todo caso no ordenados por el usuario, y a través de toda la escena. Los controles también deben ser sencillos.
- Durante la ejecución virtual del recorrido del dron, la cámara también debe moverse de forma no brusca.
- No debe requerir un tiempo de aprendizaje largo.
- Al ser el español el idioma en el que se realiza este trabajo, el prototipo también se realizará en español.
- En todo caso, debe cumplir los principios de la usabilidad. A saber, flexibilidad, consistencia, robustez, etc., de forma que pueda ser usada con efectividad, eficiencia y satisfacción.

Cabe decir que se optó por la opción de realizar un prototipo por la complejidad y el tiempo requerido, excesivos a estas alturas del proyecto para realizar una aplicación gráfica completa de estas características, capaz de realizar todo lo indicado en los requisitos de forma automática.

7.3. Unity

Para realizar el prototipo se decidió usar un motor gráfico ya que, de por sí, permitiría la fácil visualización de una escena virtual compuesta por modelos, nubes de puntos, etc. en los mismos formatos en los que se exportan los resultados de PhotoScan.

De entre los motores gráficos disponibles en el mercado, Unity fue el escogido. La razón principal fue la propia experiencia del alumno con él, proveniente de, por ejemplo, la asignatura cursada de Desarrollo de Videojuegos. Pero, también, se tuvo

en cuenta la gran sencillez y versatilidad que provee para realizar las tareas, permitiendo un prototipado muy rápido en comparación con otros motores gráficos.

Unity cuenta con tres versiones disponibles. Para este trabajo, basta con la versión más básica, Unity Personal Edition, enfocada a estudiantes y aficionados que desean explorar y empezar a trabar con esta herramienta [48]. Esta versión cuenta con todas las prestaciones básicas y necesarias del motor. Además, es gratuita mientras se cumpla la condición de no superar los 100.000\$ de ingresos brutos anuales o totales. Como no es el caso, tampoco aumentaría el coste de este proyecto.

7.4. Aspectos del desarrollo

En este apartado se mencionan diversos aspectos específicos e importantes del desarrollo del prototipo. En primer lugar, un dato para tener en cuenta es que se decidió utilizar los resultados obtenidos de la reconstrucción del segundo conjunto de imágenes, el de la finca de Torrecillas, como muestras para él. Seguidamente, se muestran otros aspectos destacables:

7.4.1. Importación de modelo 3D

El tamaño del archivo del modelo 3D procedente de PhotoScan era demasiado grande. Costaba realizar la importación a Unity y, una vez importado, era muy difícil de manejar, al menos en los equipos utilizados. A duras penas se conseguía mover la cámara a través de la escena. Los modelos contaban con demasiado nivel de detalle, demasiada cantidad de polígonos. Para reducirlo y mejorar el rendimiento en Unity, se probaron varias opciones:

La primera de ellas fue utilizar el propio PhotoScan. PhotoScan provee una herramienta para reducir la malla. Su utilización es muy simple: muestra en pantalla el número de caras actual de la malla y pide introducir un número de caras deseado (Ilustración 7.1). Cuando se introduce y se pone en marcha, intenta ajustar la malla lo máximo posible a ese número de polígonos.

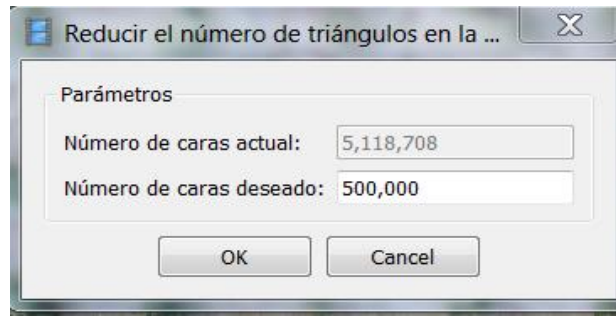


Ilustración 7.1 - Ventana que muestra PhotoScan para la utilización de la herramienta de reducción de la malla. Fuente: Elaboración propia.

En este caso, de unos 5 millones de polígonos con los que contaba la malla original, se indicó un objetivo de 500.000, una reducción del 90%. El procesamiento de estas tareas no es muy largo y el resultado puede observarse en las siguientes imágenes, que muestran detalles de la malla antes y después:

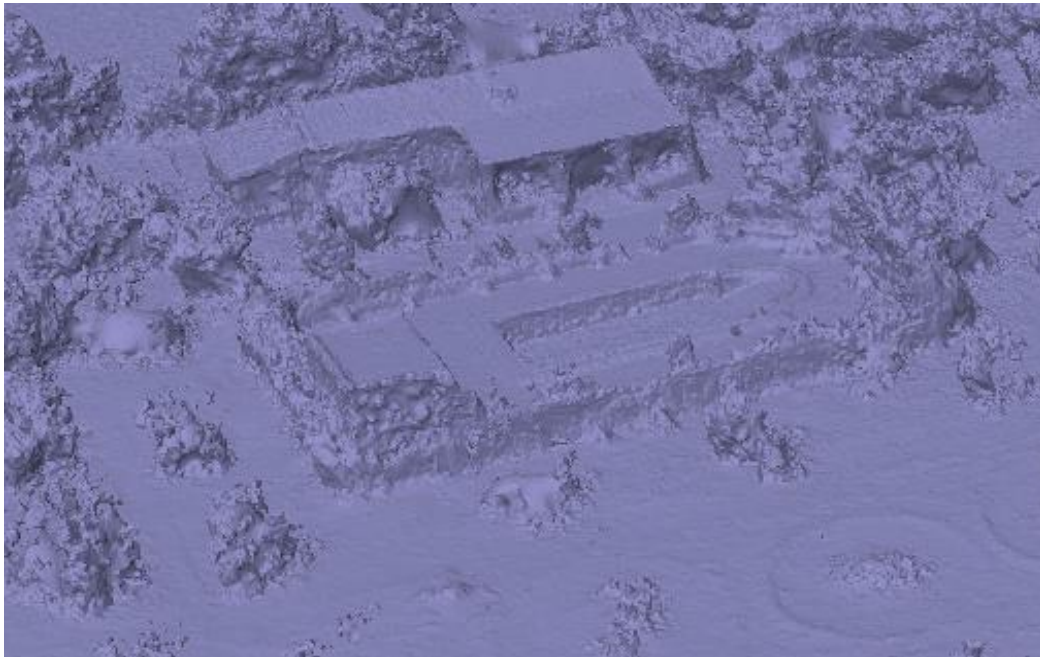


Ilustración 7.2 - Detalle de la malla antes de aplicar la reducción de polígonos.

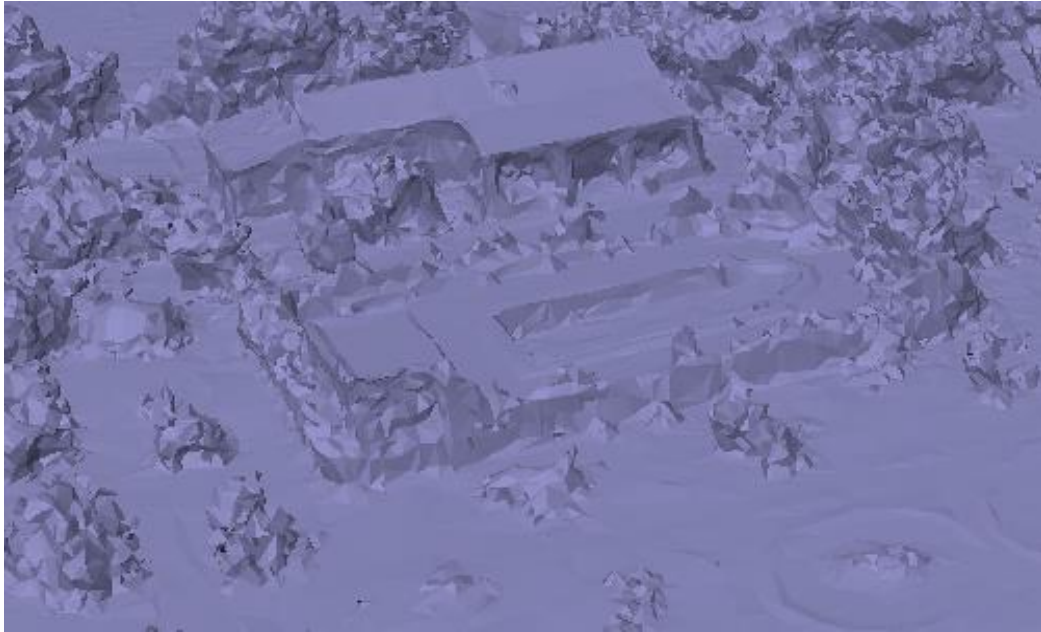


Ilustración 7.3 - Detalle de la malla después de aplicar la reducción de polígonos.

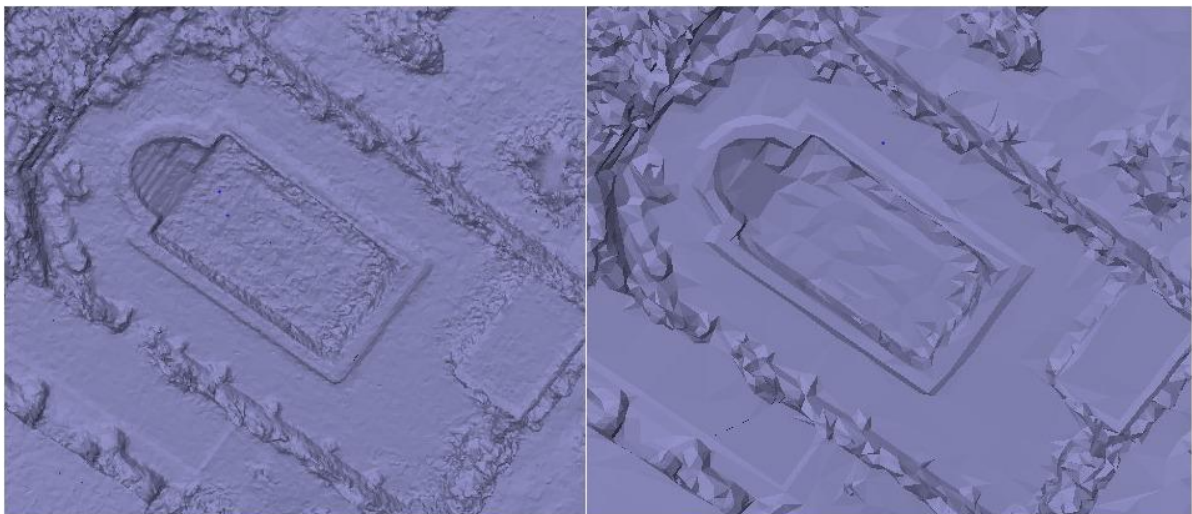


Ilustración 7.4 - Detalle distinto de la malla antes (izquierda) y después (derecha) de aplicar la reducción de polígonos. Fuente: Elaboración propia.

Se puede observar que el resultado es muy bueno. Visualmente, apenas se aprecia el procesamiento en cuanto a calidad del modelo, que es el objetivo principal. Sin embargo, fijándose en los detalles, se puede ver la gran simplificación y reducción de polígonos que se ha realizado. Se podría simplificar aún más, pero esta disminución es más que suficiente para que el modelo pueda ser perfectamente manejable dentro de Unity, por lo que se considera que no es necesario.

Para explorar otras opciones que quizá pudieran dar un resultado aún mejor, se probó la utilización del programa ZBrush. En él, gracias a la herramienta ZRemesher, se consigue también disminuir el número de polígonos que conforman una malla. Se

probó su uso con el mismo modelo mencionado anteriormente. Con los parámetros por defecto, el resultado obtenido fue una malla de 29.000 polígonos, por los 5 millones iniciales -un 99'2% de reducción-. Sin embargo, la malla obtenida era demasiado simple. Se habían perdido muchos detalles y habían surgido defectos.



Ilustración 7.5 - Vista de malla después de aplicar la reducción de polígonos con el programa ZBrush.
Fuente: Elaboración propia.

Para intentar mejorar el resultado, se probó a cambiar los parámetros de la reducción. Sin embargo, los resultados nunca fueron los esperados. Esto se debe, entre otras cosas, a que, por defecto, no es posible indicar un objetivo de polígonos mayor de 100.000, los cuales son aún demasiado pocos para un buen resultado con este tipo de superficies.

Debido a lo visto, PhotoScan fue la opción escogida para realizar la simplificación. Además, pensando en un posible usuario que requiera simplificar la malla, PhotoScan sería su opción más cómoda, ya que es el propio programa que

realiza la reconstrucción. Incluso se podría incluir la simplificación de la malla como un proceso más en el script utilizado para ello.

Con la malla simplificada, se pudo realizar la importación de forma rápida y correcta en el proyecto de Unity. Asimismo, no afectaba al rendimiento como anteriormente, y mucho menos de forma excesiva.

7.4.2. Visualización de posiciones desde las que se tomaron las fotografías

PhotoScan permite exportar en distintos formatos tanto las coordenadas geográficas reales desde las que se realizaron las fotografías, como también las mismas posiciones relativas a la escena virtual. Estas últimas serían las utilizadas en el prototipo para visualizar en la escena las posiciones desde las que el dron captó las fotografías y, así, cumplir dicho requisito funcional.

Las posiciones se exportaron en un simple formato de texto “.txt”. A pesar de que pueden ser muchas, es un formato fácil de utilizar y, para realizar el prototipo, es la mejor opción por su sencillez.

Desde Unity, en el proyecto se realizó un script “CamerasManager” para la lectura de este fichero. Además, puesto que la posición de las cámaras no es el único dato importante y que se utilizaría con respecto a ellas, se creó una clase “CameraClass” que encapsula los datos utilizados de cada una de ellas.

El script desde el que se realiza la lectura del fichero, en primer lugar, para cada cámara, toma los datos requeridos. Con estos datos, crea un nuevo objeto “CameraClass” que, más tarde, guarda en una lista para posibles necesidades futuras. A continuación, coloca un objeto, un modelo 3D -o prefab de Unity- realizado para simular una cámara, en la posición leída. De esta forma, en cada una de las posiciones virtuales desde las que se captaron las fotografías, aparecerá dibujada una nueva instancia de este objeto.

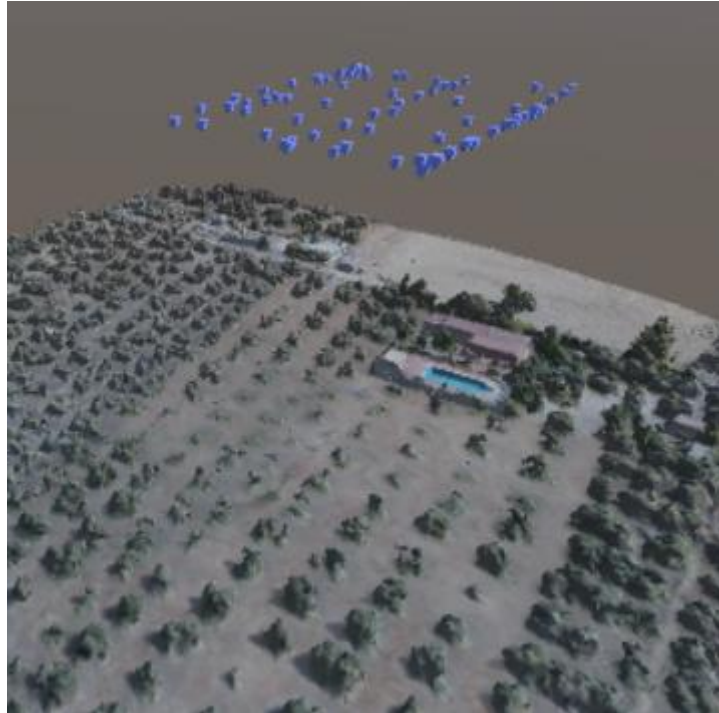


Ilustración 7.6 - Resultado de la ejecución del visualizado de posiciones virtuales desde las que se captaron las fotografías. Fuente: Elaboración propia.

7.4.3. Otros aspectos

A la hora de realizar el travelling, o recorrido virtual, del vuelo realizado por el dron, se utilizan las instancias de los objetos “CameraClass” contruídos y guardados al inicio de la ejecución. Éstos, entre otros datos, contenían las posiciones virtuales desde las que se realizó cada fotografía. Al ejecutar el recorrido virtual, se van tomando pares de posiciones por orden y se realiza una interpolación lineal en base al tiempo. Por ejemplo, si la primera posición es (0, 50, 0) y la segunda (0, 50, 10), y el tiempo indicado de recorrido entre cada posición es 4 segundos, en el segundo cero, la cámara se encontrará en la posición (0, 50, 0), en el segundo 2, por ejemplo, en la posición (0, 50, 5) y, en el segundo 4, ya en la posición (0, 50, 10).

El resultado final se encuentra en los anexos. Por un lado, a modo de proyecto de Unity. Por otro, a modo de ejecutable. Por último, también se incluye un vídeo donde se observa su funcionamiento.

Capítulo 8.

Conclusiones.

8. CONCLUSIONES

8.1. Conclusiones

La visión 3D es la forma natural que tiene el ser humano para percibir su entorno. En el presente trabajo -y, en general, en el campo de la fotogrametría aérea con drones-, se ha puesto en práctica la habilidad para resolver problemas en entornos de carácter multidisciplinar. Se unen áreas tales como la de los drones, visión por ordenador, análisis de imágenes, geomática, álgebra o estadística, para así realizar estudios de superficies o terrenos. Todo ello con el objetivo de aumentar la capacidad de estudio a las tres dimensiones y sin suponer, como en el pasado, un alto coste en tiempo y recursos.

En general, a medida que se avanzaba en este TFG, además de realizar y mostrar la utilidad de estos estudios, todos ellos han sido examinados en cada una de sus fases. Así, se ha conseguido desarrollar nuevas herramientas y prototipos que, siendo pulidos, pueden llegar a ofrecer al sector medios que automatizan, facilitan y unifican todos los procedimientos básicos necesarios en estas tareas, junto a otras utilidades.

También se han visto las posibilidades del análisis de imágenes aéreas para ampliar el estudio de la superficie. No se ha entrado en demasiado detalle, pero aun así, la capacidad de realizar análisis de índices de vegetación de forma tan sencilla ilustra la gran cantidad de posibilidades para realizar otros cálculos con otras fórmulas para estudios más complejos. Estas podrían ser usadas, por ejemplo, en el campo de la agricultura de precisión.

Los buenos resultados obtenidos demuestran que la fotogrametría aérea es una herramienta de gran utilidad en nuestros días. Además, como se ha visto, de ella derivan muchísimas vías de estudio, lo que también la convierte en un área con una gran proyección futura.

8.2. Trabajos futuros

Como trabajos futuros, en general, existen posibilidades muy variadas. En esta sección se plantean algunas de ellas:

- Desarrollar otros procesos de filtrado de imágenes que se ajusten más a las necesidades específicas del usuario para cada problema. Por ejemplo:
 - Aprovechar el dato Exif de posición GPS de cada imagen para eliminar aquellas que no se tomaran dentro de un área específica.
 - Eliminar aquellas imágenes que presenten un grado de superposición demasiado alto con respecto a otras, ya que serían prácticamente iguales entre ellas y proveerían información demasiado redundante.
- Aprovechar el modelo 3D reconstruido para ampliar el ámbito de estudio de la superficie a otros campos como los estudios cartográficos.
- Aprovechar las fotografías multiespectrales para realizar análisis más profundos de la superficie, además de los índices de vegetación. Por ejemplo, para entrar en el campo de la agricultura de precisión, la detección de aguas subterráneas, etc.
- Desarrollar completamente la aplicación que se representa con el prototipo realizado, añadiendo más funcionalidad o mejoras. Por ejemplo:
 - Incluir en ella el proceso de filtrado de imágenes a través de la interfaz.
 - La ejecución automática mediante la API de PhotoScan de las tareas de reconstrucción básicas.
 - Añadir la capacidad de ver qué fotografías concretas han influido para la reconstrucción de una zona o punto específico.
 - Añadir funcionalidad a la visualización.

Capítulo 9.

Anexo.

9. ANEXO

Junto a esta documentación, se entregan una serie de anexos:

- Todas las imágenes mostradas en este documento, ordenadas con el mismo nombre.
- Los dos conjuntos de imágenes usados en este proyecto.
- Los scripts utilizados, tanto para el filtrado de imágenes como para la ejecución en la API de PhotoScan.
- Los proyectos de PhotoScan realizados. Para abrirlos, es necesario tener instalado PhotoScan. Con el fin de poder visualizar los resultados sin utilizar PhotoScan, también se han exportado en sus formatos correspondientes, para así poder abrirlos con otros programas externos. Además, en las imágenes del anexo presentes en este documento, pueden verse capturas de pantalla de ellos.
- El proyecto de Unity con el prototipo desarrollado.
- El ejecutable del prototipo desarrollado.
- Vídeo demostración del funcionamiento del prototipo.

BIBLIOGRAFÍA

- [1] Ministerio de Fomento, «Plan Estratégico de Drones,» 2018. [En línea]. Available: <https://www.fomento.gob.es/NR/rdonlyres/7B974E30-2BD2-46E5-BEE5-26E00851A455/148411/PlanEstrategicoDrones.pdf>. [Último acceso: Julio 2018].
- [2] Wikipedia, «Fotogrametría,» [En línea]. Available: <https://es.wikipedia.org/wiki/Fotogrametr%C3%ADa>. [Último acceso: Julio 2018].
- [3] «Geomática: un neologismo con mucho mundo,» [En línea]. Available: <http://geomatice.es.com/2015/07/geomatica-un-termino-con-mucho-mundo/>. [Último acceso: Julio 2018].
- [4] V. D. Hernando, «Historia de los drones,» [En línea]. Available: <http://eldrone.es/historia-de-los-drones/>. [Último acceso: Julio 2018].
- [5] H. J. F. Rousseau, Artist, *Austrian military balloons dropping incendiary bombs on Venice*. [Art]. 1868.
- [6] Wikipedia, «Unmanned aerial vehicle,» [En línea]. Available: https://en.wikipedia.org/wiki/Unmanned_aerial_vehicle. [Último acceso: Julio 2018].
- [7] «Remote Piloted Aerial Vehicles : An Anthology,» [En línea]. Available: http://www.ctie.monash.edu.au/hargrave/rpav_home.html#top. [Último acceso: Julio 2018].
- [8] Wikipedia, «Hewitt-Sperry Automatic Airplane,» [En línea]. Available: https://en.wikipedia.org/wiki/Hewitt-Sperry_Automatic_Airplane. [Último acceso: Julio 2018].
- [9] Wikipedia, «Radioplane OQ-2,» [En línea]. Available: https://en.wikipedia.org/wiki/Radioplane_OQ-2. [Último acceso: Julio 2018].
- [10] «The Ryan Firebee: Grandfather to the Modern UAV,» [En línea]. Available: <https://gizmodo.com/the-ryan-firebee-grandfather-to-the-modern-uav-1155938222>. [Último acceso: Julio 2018].
- [11] Wikipedia, «General Atomics MQ-1 Predator,» [En línea]. Available: https://es.wikipedia.org/wiki/General_Atomics_MQ-1_Predator. [Último acceso: Julio 2018].
- [12] G. Casagrande, A. Sik y G. Szabó, *Small Flying Drones. Applications for Geographic Observation*, Springer, 2017.
- [13] M. Westoby, J. Brasington, N. Glasser, M. Hambrey y J. Reynolds, «'Structure-from-Motion' photogrammetry: A low-cost, effective tool for geoscience applications,» *Geomorphology*, vol. 179, pp. 300-314, 2012.

- [14] Wikipedia, «LIDAR,» [En línea]. Available: <https://es.wikipedia.org/wiki/LIDAR>. [Último acceso: Agosto 2018].
- [15] indeed, «Sueldos en Gestor/a de proyectos en España,» [En línea]. Available: <https://www.indeed.es/salaries/Gestor/a-de-proyectos-Salaries>. [Último acceso: Agosto 2018].
- [16] indeed, «Sueldos en Analista programador/a en España,» [En línea]. Available: <https://www.indeed.es/salaries/Analista-programador/a-Salaries>. [Último acceso: Agosto 2018].
- [17] «indeed,» Sueldos en Informatico/a en España, [En línea]. Available: <https://www.indeed.es/salaries/Informatico/a-Salaries>. [Último acceso: Agosto 2018].
- [18] eldiario.es, «De territorio sin ley a tener tres: así están regulados los drones en España,» 2014. [En línea]. Available: https://www.eldiario.es/hojaderouter/ilegales/drones-ley-regulacion-normativa-Espana_6_321677863.html. [Último acceso: Julio 2018].
- [19] dronair.es, «Nueva legislación de drones en España para 2017,» 2017. [En línea]. Available: <https://www.dronair.es/actualizacion-de-la-ley-sobre-el-uso-de-drones-en-espana>. [Último acceso: Julio 2018].
- [20] Herrero & Asociados, «Los drones y la privacidad,» [En línea]. Available: <http://www.herrerodigital.com/blog/los-drones-y-la-privacidad/>. [Último acceso: Julio 2018].
- [21] «Real Decreto 1036/2017, de 15 de diciembre,» [En línea]. Available: <https://www.boe.es/boe/dias/2017/12/29/pdfs/BOE-A-2017-15721.pdf>. [Último acceso: Julio 2018].
- [22] aerial insights, «Normativa sobre drones en España [2018],» [En línea]. Available: <http://www.aerial-insights.co/blog/normativa-drones-espana/>. [Último acceso: Agosto 2018].
- [23] E. Salamí, C. Barrado y E. Pastor, «UAV Flight Experiments Applied to the Remote Sensing of Vegetated Areas,» 2014. [En línea]. Available: <https://www.mdpi.com/2072-4292/6/11/11051ag>. [Último acceso: Julio 2018].
- [24] SkyDron, «Supervisión aérea con drones de construcciones y obra civil,» [En línea]. Available: <https://www.skydron.es/supervision-drones-construccion-obra-civil/>. [Último acceso: Agosto 2018].
- [25] «Tipos de drones,» [En línea]. Available: <http://www.xdrones.es/tipos-de-drones-clasificacion-de-drones-categorias-de-drones/>. [Último acceso: Julio 2018].
- [26] Secretaría de Comunicaciones y Transportes (México), «Plan de Vuelo,» [En línea]. Available: <https://seneam.gob.mx/scta/planvuelo.asp>. [Último acceso: Agosto 2018].

- [27] Agisoft, «Image Capture Tips: Equipment and Shooting Scenarios,» [En línea]. Available: http://www.agisoft.com/pdf/tips_and_tricks/Image%20Capture%20Tips%20-%20Equipment%20and%20Shooting%20Scenarios.pdf. [Último acceso: Julio 2018].
- [28] DJI, [En línea]. Available: <https://www.dji.com/es>. [Último acceso: Julio 2018].
- [29] DJI, «Mavic Pro,» [En línea]. Available: https://store.dji.com/es/product/mavic-pro?gclid=EAIaIQobChMI7uqYw4nJ3AIV0kPTCh0QGgWaEAAYAAAEgKGk_D_BwE. [Último acceso: Julio 2018].
- [30] Parrot, «Parrot Sequoia,» [En línea]. Available: <https://www.parrot.com/soluciones-business/profesional/parrot-sequoia>. [Último acceso: Julio 2018].
- [31] Parrot, «Configuring Sequoia,» [En línea]. Available: <https://community.parrot.com/t5/Sequoia-Knowledge-Base/Configuring-Sequoia/ta-p/137710>. [Último acceso: Julio 2018].
- [32] Pix4D, «Pix4Dcapture,» [En línea]. Available: <https://pix4d.com/product/pix4dcapture/>. [Último acceso: Julio 2018].
- [33] Pix4D, «Pix4Dmapper,» [En línea]. Available: <https://pix4d.com/product/pix4dmapper-photogrammetry-software/>. [Último acceso: Julio 2018].
- [34] «Exiftool,» [En línea]. Available: <https://www.sno.phy.queensu.ca/~phil/exiftool/>. [Último acceso: Agosto 2018].
- [35] R. Segura Sánchez, *Diapositivas de la asignatura Técnicas de Animación 3D y post-procesamiento. Tema 3.*, 2018.
- [36] elphel, «Stereo Setup,» [En línea]. Available: https://www.elphel.com/www3/stereo_setup. [Último acceso: Agosto 2018].
- [37] C. y. H. M. Holzmamann, «Measuring Distance with Mobile Phones Using Single-Camera Stereo Vision.,» *2012 32nd International Conference on Distributed Computing Systems Workshops*, pp. 88-93, 2012.
- [38] D. Lowe, «Distinctive Image Features from Scale-Invariant Keypoints,» *International Journal of Computer Vision*, vol. 60, nº 2, pp. 91-110, 2004.
- [39] P. Flores y J. Braun, «Algoritmo SIFT: fundamento te,» [En línea]. Available: <http://iie.fing.edu.uy/investigacion/grupos/gti/timag/trabajos/2011/keypoints/FundamentoSIFT.pdf>. [Último acceso: Agosto 2018].
- [40] H. Bay, T. Tyutelaars y L. Van Gool, «SURF: Speeded up Robust Features,» *Computer Vision-ECCV 2006. Lecture Notes in Computer Science*, vol. 3951, pp. 404-417, 2006.

- [41] E. Rublee, V. Rabaud, K. Konolige y G. Bradski, «ORB: an efficient alternative to SIFT or SURF,» *Proc. IEEE International Conference on Computer Vision*, vol. 13, pp. 2564-2571, 2011.
- [42] Wikipedia, «Nearest neighbor search,» [En línea]. Available: https://en.wikipedia.org/wiki/Nearest_neighbor_search. [Último acceso: Agosto 2018].
- [43] Wikipedia, «Structure from motion,» [En línea]. Available: https://es.wikipedia.org/wiki/Structure_from_motion. [Último acceso: Agosto 2018].
- [44] Agisoft, [En línea]. Available: <http://www.agisoft.com/>. [Último acceso: Agosto 2018].
- [45] Agisoft PhotoScan, «PhotoScan Python Reference,» [En línea]. Available: http://www.agisoft.com/pdf/photoscan_python_api_1_4_3.pdf. [Último acceso: Julio 2018].
- [46] Agisoft, «Agisoft Photoscan User Manual,» [En línea]. Available: http://www.agisoft.com/pdf/photoscan-pro_1_4_en.pdf. [Último acceso: Julio 2018].
- [47] Centro de Información de Recursos Naturales, «Índices de vegetación,» [En línea]. Available: <http://bibliotecadigital.ciren.cl/bitstream/handle/123456789/26389/Tema%20Indice%20de%20vegetaci%C3%B3n%20Pedro%20Mu%C3%B1oz%20A.pdf?sequence=1&isAllowed=y>. [Último acceso: Agosto 2018].
- [48] Unity, «Unity Personal,» [En línea]. Available: <https://store.unity.com/es/products/unity-personal>. [Último acceso: Agosto 2018].
- [49] «The 'Aerial Target' and 'Aerial Torpedo' in the USA,» [En línea]. Available: http://www.ctie.monash.edu.au/hargrave/rpav_usa.html. [Último acceso: Julio 2018].
- [50] «Algorithms used in Photoscan,» [En línea]. Available: <http://www.agisoft.com/forum/index.php?PHPSESSID=d5ca9b15d400e8a53af950bf0ced67f0&topic=89.msg323#msg323>. [Último acceso: Agosto 2018].