



UNIVERSIDAD
COMPLUTENSE
MADRID

Proyecto de Innovación

Convocatoria 2020/2021

Nº de proyecto: 307

Restauración de obras escultóricas mediante el empleo de Software libre

Responsable del proyecto:
Darío Lanza Vidal

Facultad de Bellas Artes

Departamento de Escultura y Formación Artística

Memoria de actividades

RESTAURACIÓN DE OBRAS ESCULTÓRICAS MEDIANTE EL EMPLEO DE SOFTWARE LIBRE

1. INTRODUCCIÓN.....	3
Resumen.....	3
Descripción del proyecto.....	3
Objetivo del proyecto y propuestas de valor del mismo.....	3
Justificación e idoneidad de los recursos necesarios.....	4
Impacto esperado e indicadores propuestos para medirlo en relación con los objetivos formulados.....	5
Viabilidad.....	5
Transferencia.....	5
Recursos humanos. Integrantes del proyecto.....	5
UNIDADES DIDÁCTICAS.....	6
2. CONSTRUCCIÓN DE UNA SUPERFICIE ROTATORIA CON ARDUINO.....	7
Diseño del mecanismo rotatorio.....	7
Programación del cerebro electrónico.....	9
3. SESIÓN DE CAPTURA FOTOGRAMÉTRICA.....	13
Preparación del plató.....	13
Ajustes de cámara.....	14
Captura fotogramétrica.....	15
4. RESTITUCIÓN FOTOGRAMÉTRICA CON MESHROOM.....	21
Organización de la interfaz de Meshroom.....	21
Proceso de trabajo.....	23
5. UNIÓN Y REESCALADO DE LAS PIEZAS EN BLENDER.....	25
6. CREACIÓN DE LOS BRAZOS CON BLENDER.....	28
Planteamiento de la reconstrucción. Período y estilo.....	28
Análisis del escaneado y su postura.....	28
Escultura digital de los brazos.....	30
7. MODELADO 3D Y ESCULTURA DIGITAL DE LA CABEZA CON BLENDER.....	35
Modelado poligonal del rostro.....	35
Escultura digital de la cabeza y cabellera.....	37
Conexión con el cuello y preparación para imprimir.....	38
8. IMPRESIÓN 3D.....	41
Electrónica.....	43
Software de impresión.....	43
Preparación del modelo para impresión.....	45
9. CONCLUSIÓN DEL PROYECTO.....	49
10. TABLA RESUMEN DE HERRAMIENTAS LIBRES Y RECURSOS ONLINE.....	52

Proyecto de Innovación y Mejora de la Calidad Docente nº 307.
 Departamento de Escultura y Formación Artística.
 Responsable del Proyecto: Darío Lanza Vidal. dlanza@ucm.es.
 Convocatoria 2020/20201

1. INTRODUCCIÓN

Resumen

La digitalización del patrimonio cultural es hoy en día uno de los ámbitos de las denominadas humanidades digitales que se encuentra en mayor expansión. Tanto con fines de investigación como para la restauración, conservación o difusión de bienes culturales, son innumerables los contextos en los que esta disciplina resulta de utilidad. Sin embargo suele percibirse que este tipo de tareas exigen un importante desembolso económico en equipamiento y software informático, quedando su uso al alcance únicamente de grandes instituciones. Con el presente proyecto hemos intentado poner en cuestión esta creencia desarrollando un proyecto completo de restauración escultórica mediante únicamente software y hardware libre o gratuito. Desde el inicio del proyecto, con la creación de una mesa rotatoria automatizada para facilitar el escaneado, hasta la impresión final de las piezas faltantes en la escultura, pasando por la captura de las imágenes, la restitución fotogramétrica, el procesamiento del modelo y la creación de los fragmentos perdidos, todo ello ha sido realizado con la doble premisa de incurrir en un coste económico próximo a cero y en que la calidad y precisión de los resultados sea completamente profesional. La presente memoria relata la experiencia y permite constatar el éxito de nuestra propuesta.

Palabras clave: Software libre; Escultura; Fotogrametría; Modelado 3D; Escultura digital; Impresión 3D.

Descripción del proyecto

Las tecnologías digitales, que hasta la fecha parecían no hallar conexión con disciplinas tradicionales como la escultura o la restauración del patrimonio, están experimentando en los últimos tiempos desarrollos de gran importancia en estos ámbitos. Proyectos como la reciente digitalización de las piezas encontradas en la tumba de Tutankamón, la restauración de obras escultóricas de Antonio Canova mediante escaneado e impresión 3D, las versiones digitales de las colecciones del Museo Arqueológico Nacional con vistas a su difusión online, las cada vez más habituales visitas virtuales a museos y entornos patrimoniales, o las experiencias de realidad virtual y realidad aumentada están demostrando que las tecnologías digitales pueden jugar un papel de extraordinaria relevancia en estos contextos.

Sin embargo, vemos con frecuencia que el alumno percibe un abismo insalvable para aproximarse a estas disciplinas al asumir que el trabajo en dicho sector exige inevitablemente un gran desembolso económico en hardware y software específicos, como costosos escáneres láser o programas con licencias prohibitivas. Con el presente proyecto pretendemos salvar dicho espacio y aproximar al alumno a estas disciplinas demostrando la posibilidad de desarrollar la restauración completa de una obra escultórica fragmentada empleando herramientas digitales libres y gratuitas.

Surgido como continuación de otros dos PIMCDs sobre las posibilidades docentes y creativas del software libre, los miembros participantes en este proyecto, todos ellos expertos en materias de tecnologías digitales, realizarán un proyecto completo de restauración de una obra escultórica fragmentada, desde la captura fotográfica, la restitución fotogramétrica, la construcción 3D de las piezas faltantes y la obtención física de dichos elementos mediante impresión 3D, empleando durante el proceso únicamente softwares libres o gratuitos y produciendo, al mismo tiempo que el desarrollo del proyecto, una memoria de trabajo que servirá a alumnos y profesores como manual con el que aprender tanto sobre la creación escultórica digital como sobre el uso práctico de los softwares empleados. Ambos elementos, la pieza final restaurada y la memoria/manual, se publicarán en abierto para garantizar la mayor difusión de los desarrollos y metodologías investigadas durante el proyecto.

Un proyecto, en resumen, que busca dar difusión a las tecnologías libres como herramientas de gran interés para el artista actual, que ayuda a aproximar las herramientas digitales a contextos artísticos que hasta la fecha no habían encontrado conexión con el entorno digital, que proporciona material docente para profesores de asignaturas tecnológicas, que otorga un gran valor tanto para el escultor contemporáneo como para el restaurador de patrimonio sin requerir desembolso económico, y que prolonga su vocación educativa con la publicación de una memoria que servirá como manual o guía práctica para alumnos y futuros profesionales.

Objetivo del proyecto y propuestas de valor del mismo

En base a los programas del Grado de Bellas Artes incluidos en asignaturas de perfil tecnológico, como "Tecnologías Digitales", "Modelización 3D" o "Modelado 2D/3D para videojuegos" del Grado en Diseño de Videojuegos, asignaturas que son impartidas por los miembros del presente equipo, este proyecto de innovación educativa propone los siguientes objetivos:

Objetivos Generales:

- Introducir el uso y manejo de herramientas de escultura digital y creación escultórica en 3D mediante software libre, así como su aplicación para la creación artística y la preservación del patrimonio.
- Comprender y utilizar los fundamentos de las tecnologías digitales, estrategias, métodos y procesos de trabajo.
- Adecuar la tecnología disponible a los objetivos en cada proceso de diseño y creación.
- Aprender técnicas digitales por medio de software libre aplicadas, desde la captura fotogramétrica, la construcción 3D hasta la obtención física de la pieza final mediante impresión 3D.
- Capacitar al alumno para adquirir conocimientos sobre la escultura digital y su aplicación artística profesional.
- Capacitar al estudiante para la resolución de problemas de forma autónoma, creativa e innovadora.
- Fomentar la investigación y la experimentación a través de la aplicación de la tecnología digital, con el uso del software libre.
- Aplicar profesionalmente tecnologías específicas.
- Utilizar las herramientas apropiadas para los lenguajes artísticos propios.

Objetivos Específicos:

- Establecer reuniones en las que concretar los objetivos generales, en función de las particularidades técnicas de cada uno de los programas.
- Realizar unidades didácticas de los programas de software libre que vayamos trabajando, en base a un enfoque práctico. Investigaremos las capacidades de hardware libre como Arduino y programas como Meshroom para la restitución fotogramétrica, trabajaremos con Blender en el modelado y escultura digital de las piezas faltantes y finalmente utilizaremos Cura para impresión 3D de los fragmentos digitales.
- Ofrecer el resultado como una pieza 3D que pueda ser consultada online libremente desde una página web.
- Documentar, recopilar, y actualizar toda la información online de los software investigados en la forma de una memoria escrita de libre acceso que contenga las unidades didácticas con los casos prácticos realizados a través de los programas de software libre empleados. Se pretende compartir los resultados no sólo con la comunidad educativa sino también con el conjunto de la sociedad digital, tal y como plantea la filosofía del software libre en su interés por compartir conocimiento.
- Realizar una tabla o glosario de las herramientas de acceso libre sobre fotogrametría, modelado 3D, escultura digital e impresión 3D, así como de usos y aplicaciones de bibliotecas de recursos online libres.

Justificación e idoneidad de los recursos necesarios

Los miembros que componen el presente grupo de trabajo están relacionados con las áreas de conocimiento sobre las que este proyecto va a tratar y sus principales líneas de investigación se hallan comprometidas con la innovación y la creación tecnológica para el ámbito artístico. Este proyecto se plantea como continuación del PIMCD 157 de 2019/20, "Diseño y creación de videojuegos mediante herramientas de software libre" y se presenta con el propósito de profundizar y continuar dicha investigación sobre las posibilidades artísticas y docentes del software libre.

Los cinco miembros del equipo pertenecen a la Facultad de Bellas Artes de la UCM. Darío Lanza Vidal, Pablo de Arriba del Amo y Santiago Tena Morales son personal PDI en la Facultad de Bellas Artes, a quienes se suman Iria Groba Martín y Miguel Pozas Pérez como exalumnos y becarios de colaboración del Departamento de Escultura y Formación Artística.

El responsable del proyecto, Darío Lanza Vidal, es PDI del Departamento de Escultura y Formación Artística, experto en creación artística contemporánea a través de tecnología 3D. Ha participado en el desarrollo de software para el renderizado de imágenes fotorrealistas y trabajado como profesional reconocido en rendering en la empresa Next Limit, desarrollando softwares de efectos especiales para la industria cinematográfica.

Pablo de Arriba del Amo ha sido Director del Departamento de Escultura y Formación Artística, imparte las asignaturas de "Construcción y representación" y "Estrategias Artísticas. Escultura", y ha participado en el PIMCD "Aplicación de la tecnología 3D a la conservación de arte contemporáneo como solución al robo y desaparición de fragmentos de esculturas. Caso de estudio: Los Portadores de la Antorcha" dirigido por Sonia Santos Gómez.

Santiago Tena Morales ha impartido las asignaturas de "Modelización 3D", "Modelado 2D/3D para videojuegos", "Tecnologías digitales", "Estrategias artísticas" y "Producción artística escultórica". Tanto Darío Lanza, responsable del grupo, como Santiago Tena cuentan con evaluaciones docentes Positivas, Muy Positivas y Excelentes en los últimos cursos.

Estas circunstancias hacen a este equipo idóneo para el desarrollo del presente proyecto, tanto por su participación en anteriores PIMCDs de temática afin, su trayectoria docente en materias de creación escultórica y tecnología, su experiencia profesional e investigadora en dichas áreas, como por su apoyo a las herramientas libres y recursos en abierto como vía para la universalización de la formación.

Impacto esperado e indicadores propuestos para medirlo en relación con los objetivos formulados

Este proyecto de innovación docente pretende potenciar la calidad y diversidad de la información que se ofrece al alumno.

Al trabajar con software libre, el coste tecnológico es prácticamente cero. Es de dominio público y su descarga se inserta dentro de la legalidad. Permite al usuario utilizarlo libremente, mejorarlo, modificarlo y adaptarlo a sus necesidades. Presenta la ventaja de no requerir el pago de una licencia por cada computadora en que se instale, lo que permite al centro educativo reducir el coste del material tecnológico en las aulas al tiempo que a los estudiantes descubrir y utilizar herramientas de última generación a coste cero.

Viabilidad

La viabilidad del proyecto queda justificada por el propósito de cumplir las actividades enunciadas, que incluyen la investigación en las herramientas digitales libres disponibles en la actualidad, la realización de un proyecto práctico de restauración escultórica y el desarrollo de unidades docentes como vehículo para la difusión de la metodología empleada y sus resultados.

Transferencia

El presente proyecto busca incrementar la difusión del conocimiento en torno al software libre y en concreto sobre las herramientas de creación escultórica digital, una disciplina de gran interés tanto para los alumnos interesados en la creación artística contemporánea como para los dedicados a la conservación y preservación del patrimonio.

Una vez explorado el potencial de dichas herramientas libres en el citado contexto de la restauración de obras escultóricas, el equipo investigador ha redactado la presente memoria que será publicada a modo de manual cuya vocación es convertirse en un interesante recurso de estudio tanto para alumnos como para profesores de asignaturas tecnológicas interesados en impartir este tipo de docencia.

Así, la transferencia de conocimiento de este proyecto queda garantizada con la publicación de la citada memoria como recurso libre, maximizando así la difusión de los hallazgos obtenidos por el equipo investigador.

Además, se pretende establecer colaboraciones con otros centros de innovación docente centrados en el arte y la tecnología, tanto nacionales como internacionales, incrementando aún más la visibilidad de nuestro trabajo.

Recursos humanos. Integrantes del proyecto

- Darío Lanza Vidal: dlanza@ucm.es
- Pablo de Arriba del Amo: pablodearriba@art.ucm.es
- Santiago Tena Morales: santiagotena@ucm.es
- Iria Groba Martín: iriama01@ucm.es
- Miguel Pozas Pérez: migpozas@ucm.es

UNIDADES DIDÁCTICAS

2. CONSTRUCCIÓN DE UNA SUPERFICIE ROTATORIA CON ARDUINO

Pablo de Arriba del Amo
Identificador ORCID: 0000-0002-8903-8472

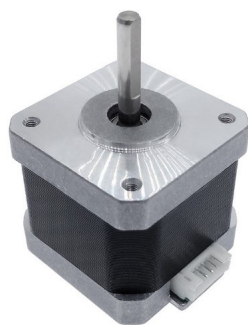
A la hora de abordar la captura de imágenes para generar un modelo digital mediante el proceso de fotogrametría hay que tener en cuenta las características específicas del motivo sobre el que se pretende trabajar, ya que podemos encontrar situaciones muy distintas derivadas de la morfología del propio objeto, sus dimensiones, sus particularidades materiales y cromáticas, y cuestiones como su accesibilidad o la posibilidad de ser manipulados para la realización de las tomas. Estas consideraciones obligan a adoptar estrategias individualizadas para lograr aportar al software un conjunto de datos homogéneo y de la mejor calidad.

La estrategia más utilizada en fotogrametría para la captura de imágenes de objetos cercanos propone la realización de una serie de bucles a distintas alturas en torno al modelo, donde éste se mantiene inmóvil y es la cámara la que se desplaza capturando los detalles morfológicos, texturales y cromáticos de la superficie, manteniendo cierto solapamiento entre las imágenes y la mayor homogeneidad posible en cuanto a condiciones de iluminación, distancia focal, exposición y apertura de diafragma. Lograr esa homogeneidad es en muchas situaciones complejo debido a que el propio desplazamiento de la cámara y el fotógrafo introduce siempre alteraciones en las condiciones de iluminación que resultan difícilmente evitables. Una alternativa que logra preservar inalteradas las condiciones lumínicas durante la captura es mantener la cámara y el sistema de iluminación estáticos durante cada bucle y que sea el objeto el que gire frente a la cámara. Si bien esta estrategia de cámara estática facilita el proceso, permite una mayor automatización y mejora la calidad de las imágenes capturadas, no es menos cierto que puede conducir a errores al software de restitución fotogramétrica. Ello es debido a que la mayoría de programas de restitución fotogramétrica se basan en el algoritmo conocido como *Structure from Motion* (SfM), el cual asume que el objeto se encuentra permanentemente estático y por tanto toda discrepancia observable entre imágenes es debida a los cambios en la posición de la cámara. Cuando, por el contrario, es la cámara la que permanece estática, el programa puede detectar incongruencias entre la posición de objetos estáticos del fondo en relación con un objeto central en movimiento, conduciéndole a una interpretación errónea de la forma o cuando menos a desechar todas las imágenes en que detecte dichas discrepancias. La solución a este problema pasa por eliminar de las capturas los datos relativos al fondo, bien mediante la aplicación de máscaras en un post-procesado de las imágenes, o bien mediante la instalación de un fondo neutro suficientemente alejado del modelo como para que el software no detecte detalles relevantes en su superficie y sea, por tanto, automáticamente ignorado.

Para la realización de este proyecto hemos optado por la estrategia descrita de mantener la cámara estática y rotar el modelo frente a ella, lo que nos ha obligado a diseñar y construir una superficie rotatoria que pudiera soportar el peso de la escultura, con un mecanismo programable para poder sincronizarla con los disparos de la cámara y, en caso necesario, con el sistema de iluminación de la escena, permitiendo así la automatización de todo el proceso, y a instalar un fondo neutro que eliminase los mencionados errores en el proceso fotogramétrico. Los criterios seguidos para diseñar la plataforma rotatoria han sido los siguientes: en primer lugar debía estar construido mediante hardware libre, y por lo tanto de libre acceso y susceptible a cualquier tipo de adaptación, cambio o mejora que considerásemos necesario para el proyecto. Por otra parte, se siguió la ética de la cultura *Do It Yourself* (DIY), o “hazlo tú mismo”, imponiéndonos el requisito de que la construcción fuera asequible en cuanto a costes y facilidad de realización. En el aspecto técnico el criterio primordial habría de ser lograr un sistema fácil de usar dentro del entorno de un estudio de escultura o de conservación y restauración del patrimonio, al mismo tiempo que capaz de proporcionar un conjunto de capturas fotográficas de la mayor calidad y homogeneidad.

Diseño del mecanismo rotatorio

El mecanismo giratorio de la mesa se construyó empleando un motor paso-a-paso, o *stepper*, y una reductora de engranajes. Los motores paso-a-paso permiten controlar la posición del motor con precisión y repetividad, permitiendo establecer la cantidad de grados a rotar en cada movimiento (fig. 1.1).

Fig. 1.1. *Stepper* o motor paso-a-paso

Para lograr una mayor precisión y suavidad en el movimiento se incorporó una transmisión mediante un sistema reductor que aumenta el torque o fuerza del par motor sobre el eje principal de la plataforma, logrando incrementar la potencia transmitida al eje principal para evitar posibles bloqueos o pérdidas de pasos por exceso de carga sobre la plataforma, pero a costa de una cierta reducción en la velocidad de giro. Afortunadamente en nuestro proyecto esta disminución de la velocidad de rotación no conlleva una ralentización en el ritmo de trabajo, ya que en el propio proceso de captura va a ser necesario introducir, como se detallará más adelante, un breve lapso de tiempo entre cada disparo para estabilizar el sistema, eliminar cualquier vibración en la escultura provocada por el movimiento de rotación y para dar tiempo a la escritura del archivo fotográfico en la memoria de la cámara. Es más, mayor velocidad de rotación implicaría también una mayor inercia, que dificultaría la estabilización del conjunto y la eliminación de vibraciones en la escultura.

En el mercado se encuentran disponibles diversos mecanismos de reducción, dependiendo de la posición del motor con relación al eje principal de la plataforma, la relación de reducción deseada, el espacio disponible para alojar el sistema o el torque de salida requerido (fig. 1.2).



Fig. 1.2. Sistema de engranajes reductores (arriba izqda). Reductora coaxial (arriba dcha). Reducción mediante husillo (abajo izqda). Reductora mediante poleas y correas dentadas (abajo dcha)

Durante el presente proyecto se construyeron dos plataformas: una de 30cm de diámetro equipada con motor Nema 17 y un sistema reductor de engranajes impresos en plástico ABS, y otra mayor, con un diámetro de

60cm, movida por un motor Nema 24 y con reducción mediante poleas y correas dentadas. Otra de las ventajas de disminuir la velocidad de rotación es que bajas velocidades provocan menores desgastes en las piezas, lo que, en el caso del sistema de engranajes, nos habilita la posibilidad de imprimirlos en plástico ABS o incluso PLA, siempre y cuando su diseño sea lo suficientemente robusto como para soportar las tensiones que experimentan las piezas durante el giro. La utilización de elementos impresos en 3D se alinea completamente con los principios de este proyecto y hace que su realización resulte no sólo asequible sino fácilmente personalizable, al poder redimensionarse para estructuras de cualquier otro tamaño mediante sencillas adaptaciones en los modelos 3D (fig. 1.3).

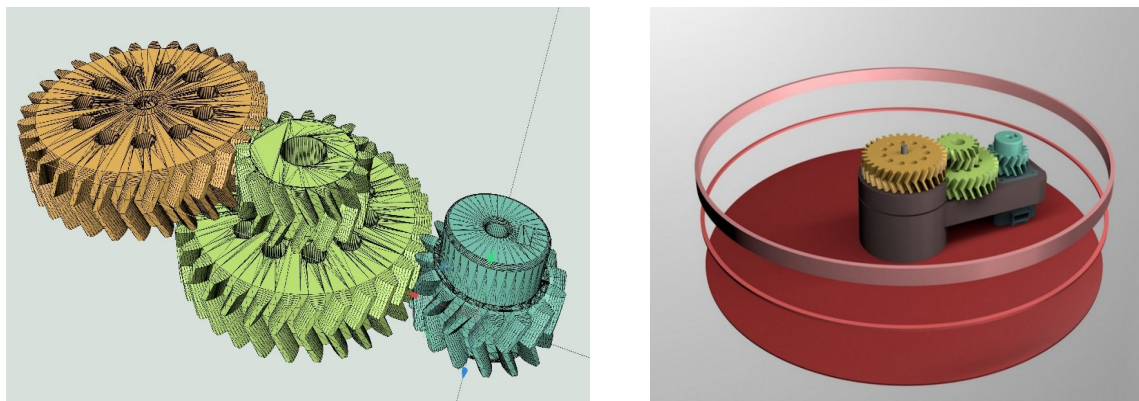


Fig. 1.3. Modelo 3D de los engranajes reductores antes de su impresión en filamento ABS (izqda). Instalación del mecanismo reductor en la plataforma (dcha).

Programación del cerebro electrónico

La automatización y sincronización de la superficie rotatoria se realizaron mediante una placa Arduino, proyecto de desarrollo de software y hardware libre, flexible y fácil de utilizar, con una extensa y activa comunidad internacional de usuarios que diseña y manufactura placas controladoras programables que permiten construir dispositivos digitales interactivos capaces de detectar y/o controlar elementos del mundo real. El sistema se organiza en torno a una placa electrónica que permite conectar una gran variedad de periféricos (como sensores de luz, de movimiento, de temperatura, interruptores, potenciómetros, motores servos o luces LED) a las entradas y salidas de un microprocesador que puede ser programado (fig. 1.4 izqda). Para su libre programación por parte del usuario, Arduino ofrece el entorno de programación Arduino IDE (acrónimo de “entorno de desarrollo integrado”), con el que sin necesidad de conocimientos avanzados en programación se puede dar a esta placa un sinfín de utilidades prácticas (fig. 1.4 dcha).



Fig. 1.4. Placa controladora Arduino Uno (izqda). Entorno de trabajo integrado Arduino IDE (dcha)

Para el cerebro electrónico de nuestra plataforma utilizamos una placa Arduino Uno REV3, el modelo más básico y económico de esta serie pero que dispone de suficiente memoria y conectividad para permitirnos el control de las plataformas. Para trasladar los impulsos de la placa controladora al motor paso-a-paso se utilizó el driver Pololu DRV8825, que soporta suficiente intensidad como para accionar el motor Nema 17 e incluso el Nema 24. Al circuito se añadieron pulsadores que nos permiten seleccionar entre varias subrutinas y reiniciar el sistema, relés que funcionan como interruptores para permitirnos abrir o cerrar otros circuitos independientes, como la alimentación de 12v o el disparador de la cámara, y varias luces LED que informan sobre el estado del sistema y permiten identificar la operación que está en ejecución en un determinado momento, como por ejemplo el disparo de la cámara o el final de cada bucle (fig. 1.5).

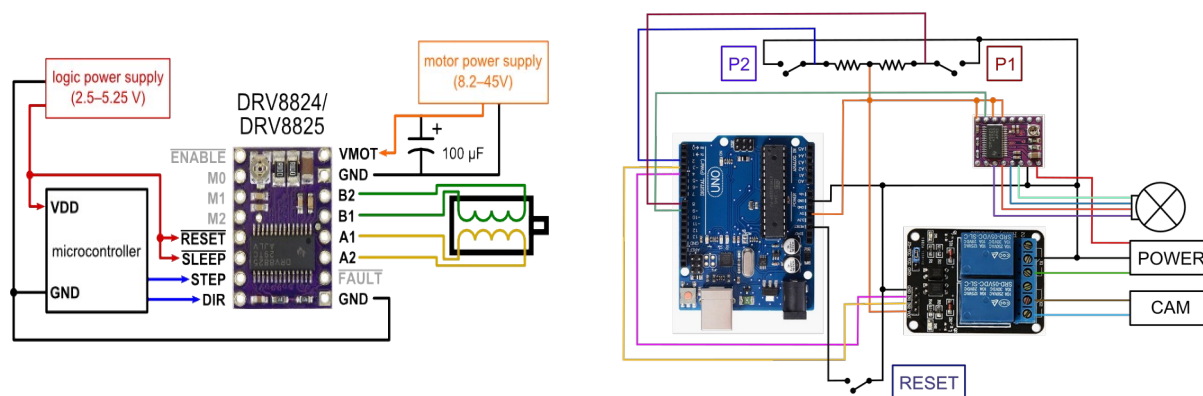


Fig. 1.5. Conexión de un driver DRV8824 al stepper y a la placa controladora (izqda). Cableado del conjunto de la electrónica: Arduino Uno, driver Pololu DRV8824, bloque de dos relés, pulsadores P1, P2 y Reset, fuente de alimentación y conexión al stepper y la cámara (dcha)

La placa Arduino Uno permite controlar todos estos dispositivos, pero la programación de su comportamiento debe ser diseñada e introducida por el usuario mediante el entorno de programación Arduino IDE. Aquí el usuario establece las variables, los datos, las funciones y las decisiones que la placa posteriormente tomará cuando esté en funcionamiento. La función principal de nuestra plataforma es la que ordena la rotación del motor paso-a-paso y el ángulo rotado, determinando el número de tomas que se realizarán en cada bucle, dato que estará relacionado con el tamaño del objeto y el grado de solapamiento o superposición entre dos imágenes consecutivas. En nuestro código establecemos dos funciones diferentes, que podemos elegir mediante sendos pulsadores: en la primera se asignan 89 pasos con un total de 114 ciclos, dando como resultado 115 imágenes por bucle; en la segunda opción los pasos serán 178, con 57 ciclos y un conjunto de 58 tomas.

Además establecemos una serie de retardos, empleando la función *delay*, con los que provocamos pequeñas pausas entre algunas operaciones para permitir que tanto la electrónica como la mecánica tengan tiempo suficiente como para completar las instrucciones que se van enviando. Como ya hemos apuntado, un aspecto importante a tener en cuenta en las capturas fotogramétricas realizadas mediante esta estrategia está relacionado con el propio movimiento de rotación de la plataforma y con ello del modelo a capturar, ya que debe garantizarse que en el momento del disparo el objeto se encuentre completamente estático y libre de vibraciones, y además ha de garantizarse el tiempo necesario para que la cámara grabe en la tarjeta los pesados archivos RAW que se van generando.

Gracias a la facilidad de programación de la placa Arduino, el número de pasos y los retardos entre las diversas operaciones son fácilmente modificables por el usuario, quien deberá personalizarlos en cada proyecto fotogramétrico en base principalmente a:

- número de tomas que deseen realizarse por vuelta,
- solapamiento deseado entre fotografías,
- relación de reducción de la transmisión,
- tiempo de estabilización del conjunto y de escritura en la tarjeta.

De cara a facilitar la recreación de este proyecto por parte de cualquier usuario, proporcionamos aquí el código con el que hemos programado nuestra plataforma rotatoria, código que puede ser muy fácilmente modificado para adaptarlo a condiciones diferentes.

```

/*Control de un motor paso a paso mediante un driver Pololu A4988 o DRV8825 con Arduino.
 * El objetivo de este código es el de controlar el movimiento de un motor sincronizando el giro
 de una plataforma con el disparo de una cámara, por medio de Arduino como microcontrolador.
 * Este circuito necesita el uso de una fuente de alimentación de 12V externa, pulsadores para
 activar los ciclos y relés para activar el disparador y la fuente de 12V
 * Habrá que definir el valor de la tensión para lograr que el motor o el driver no se sobrecalienten, para ello es preciso regular el Pololu mediante el potenciómetro que lleva incorporado
 el driver.
 * Se usarán pulsadores para el avance.
 * Cuando se hayan completado los pasos indicados se activará un relé para control de obturador
 de la cámara.
 * Quedará a la espera de que se pulse el botón Avance.
 * El motor siempre funcionará en la misma dirección.
 */

#define Avance1_PIN 2
#define Avance2_PIN 7
#define Avance_LED 8
#define Cámara1_LED 6
#define Cámara2_LED 12
#define Cámara_PIN 3
#define Steps_PIN 9
#define Alimentación_PIN 4

void setup() {
  pinMode(2, INPUT);
  pinMode(7, INPUT);
  pinMode(3, OUTPUT);
  pinMode(9, OUTPUT);
  pinMode(4, OUTPUT);
  pinMode(8, OUTPUT);
  pinMode(6, OUTPUT);
  pinMode(12, OUTPUT);
}

void loop() {
  digitalWrite(4, HIGH);
  digitalWrite(3, HIGH);
  digitalWrite(6, LOW);
  digitalWrite(8, LOW);
  digitalWrite(12, LOW);
  if (digitalRead(2) == 0) { //Al pulsar Avance se moverá 89 pasos que equivale a unos 3°
    digitalWrite(4, LOW); //Activa los 12 v de alimentación del driver
    digitalWrite(8, HIGH);
    delay(2000); //Retardo para estabilizar la tensión
    for (int h=0; h <= 114; h++){
      for (int i=0; i <= 89; i++){
        digitalWrite(9, HIGH);
        delay(3);
        digitalWrite(9, LOW);
        delay(3);
      }
    }
    delay(2000); // Retardo para que establezca la plataforma
    digitalWrite(3, LOW); // Activación del relé del obturador
  }
}

```

```
digitalWrite(6, HIGH);
delay(1000);           // Retardo después de disparar la cámara
digitalWrite(3, HIGH);
digitalWrite(6, LOW);
delay(1000);
}                       // Fin de ciclo de 3°.
digitalWrite(4, HIGH); // Desactiva la alimentación de 12V.
digitalWrite(8, LOW);
}
if (digitalRead(7) == 0) { //Al pulsar Avance se moverá 178 pasos que equivale a unos 6°
digitalWrite(4, LOW);     //Activa los 12 v de alimentación del driver
digitalWrite(8, HIGH);
delay(2000);             //Retardo para estabilizar la tensión
for (int h=0; h <= 57; h++){
for (int i=0; i <= 178; i++){
digitalWrite(9, HIGH);
delay(3);
digitalWrite(9, LOW);
delay(3);
}
delay(2000);            // Retardo para que establezca la plataforma
digitalWrite(3, LOW);  // Activación del relé del obturador
digitalWrite(12, HIGH);
delay(1000);          // Retardo después de disparar la cámara
digitalWrite(3, HIGH);
digitalWrite(12, LOW);
delay(1000);
}                       // Fin de ciclo de 6°.
digitalWrite(4, HIGH); // Desactiva la alimentación de 12V.
digitalWrite(8, LOW);
}
}
```

Fig. 1.6. Código para programar la plataforma rotatoria

3. SESIÓN DE CAPTURA FOTOGRAMÉTRICA

Iria Groba Martín

Identificador ORCID: 0000-0002-2172-0993

Preparación del plató

Tal y como hemos descrito en el capítulo anterior, para este proyecto se optó por la estrategia de cámara estática y plataforma rotatoria que hace imprescindible la instalación de un fondo neutro para evitar posteriores errores en el software de restitución fotogramétrica al identificar fotografías en las que el modelo varía de imagen a imagen mientras detecta elementos en el fondo que permanecen estáticos, lo que suele confundir a este tipo de algoritmos de posicionamiento. Como la superficie de la pieza a digitalizar era de tonos claros se eligió un fondo negro que garantizaba el máximo contraste entre figura y fondo y permitía aislar fácilmente la escultura a posteriori. Este fondo se extendió hacia el suelo y el techo, asegurando su presencia también en las fotografías picadas y contrapicadas. Para reforzar esta separación entre escultura y fondo también se pintó de negro la superficie rotatoria.

La iluminación se realizó mediante dos focos próximos a la cámara de modo que se impedía la aparición de sombras proyectadas desde el punto de vista de ésta. La altura de estos focos se fue variando gradualmente durante la toma fotográfica manteniendo constante su posición respecto a la cámara a medida que ésta se iba subiendo y bajando en cada bucle (fig. 3.1).

Finalmente se instaló la cámara en un trípode capaz de hacerla subir y bajar verticalmente para registrar los diferentes bucles en torno al modelo, e inclinarla hacia arriba y hacia abajo para tomar vistas desde angulaciones picadas y contrapicadas. La cámara se conectó por medio de un cable a la placa Arduino, la cual orquestaba la rotación de la plataforma con el disparo de las fotografías.

Debido a que la escultura se encuentra fragmentada fue necesario realizar capturas fotogramétricas independientes para cada uno de los cuatro fragmentos: cuerpo, brazo derecho, pierna y pedestal.

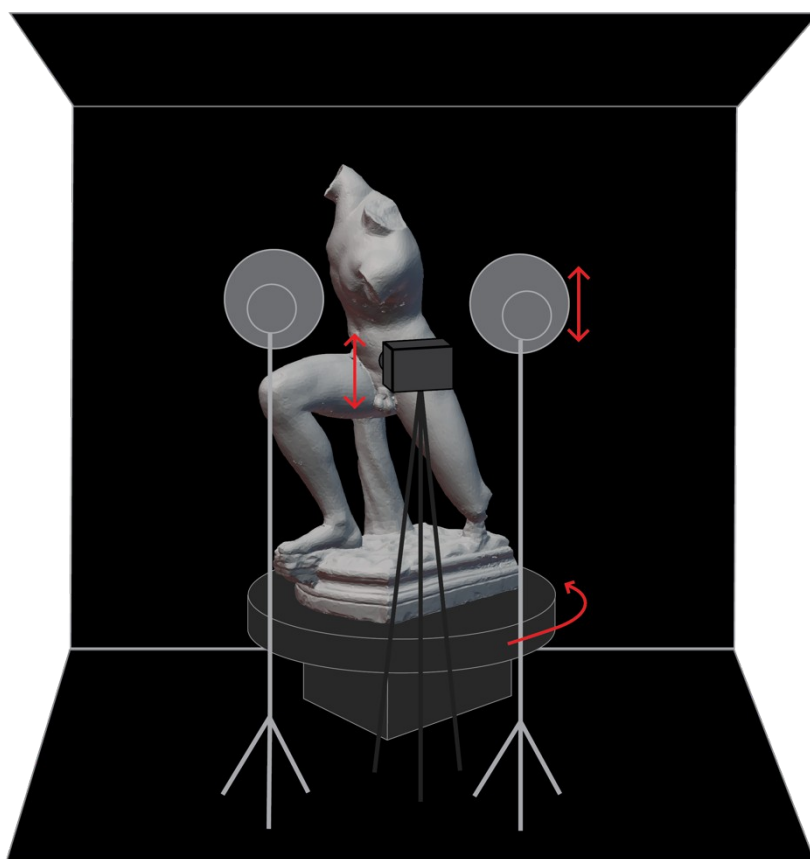


Fig. 3.1. Esquema del plató en el que puede observarse la disposición de la escultura, la plataforma, el fondo, los focos, la cámara y el trípode

Ajustes de cámara

Para las tomas fotográficas se empleó una Canon EOS 5D Mark II con sensor *full frame*. Todos los ajustes se mantuvieron manuales. Como los programas de reconstrucción fotogramétrica sólo son capaces de tomar información de las zonas de la imagen que se encuentran perfectamente enfocadas, ampliar la profundidad de campo resulta una medida interesante para incrementar la información útil en cada fotografía. En nuestro caso mantuvimos una apertura de diafragma de F16.

Generalmente, el tiempo de exposición varía dependiendo de las condiciones lumínicas. Habitualmente, cerrar mucho el diafragma tiene como consecuencia largos tiempos de exposición que requieren el uso de trípode para evitar trepidación y que las fotografías salgan movidas. En este caso, la gran cantidad de luz proporcionada por los focos permitió mantener una velocidad de obturación de 1/60 segundos en la mayoría de imágenes, aunque para algunas de las tomas fue necesario aumentarla hasta 1/13 segundos. Aquí el trípode desempeñó un papel fundamental al evitar que estas últimas saliesen movidas.

Para mejorar la calidad de las imágenes se empleó una sensibilidad lo más baja posible, ISO 200. Las fotografías se tomaron en formato RAW, que nos proporciona una mayor profundidad de color y evita cualquier compresión y pérdida de calidad.

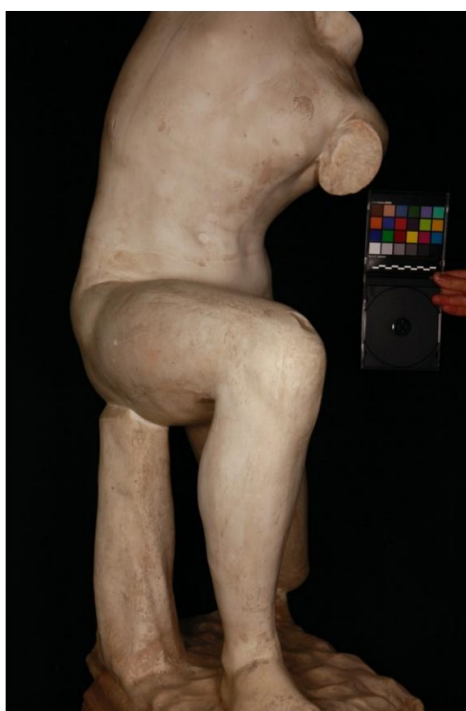


Fig. 3.2. Uso de la carta de color

Para establecer el balance de blancos, se hizo uso de una carta de color (fig. 3.2). La distancia focal se ajustó al principio de cada toma, enfocando las zonas de la escultura que se encontraban a la altura del centro de la superficie rotatoria.

Aunque la mayoría de ajustes de cámara se mantuvieron constantes, algunos parámetros como la velocidad de obturación o la distancia focal variaron ligeramente entre tomas. La fig. 3.3 muestra un resumen de los ajustes de cámara empleados en cada una de las capturas.

Pieza	Punto de vista	Formato de archivo	Espacio de color	ISO	Diafragma	Velocidad de obturación	Distancia focal
Cuerpo entero	frontal (3 tomas)	RAW	RGB	200	F16	1/60	46 mm

	picado (3 tomas)	RAW	RGB	200	F16	1/60	48 mm
	contrapicado (1 toma)	RAW	RGB	200	F16	1/60	51 mm
Brazo	frontal posición 1	RAW	RGB	200	F16	1/60	48 mm
	picado posición 1	RAW	RGB	200	F16	1/60	48 mm
	frontal posición 2	RAW	RGB	200	F16	1/60	48 mm
	picado posición 2	RAW	RGB	200	F16	1/60	48 mm
Pie	frontal posición 1	RAW	RGB	200	F16	1/25	50 mm
	picado posición 1	RAW	RGB	200	F16	1/25	50 mm
	contrapicado posición 1	RAW	RGB	200	F16	1/25	50 mm
	frontal posición 2	RAW	RGB	200	F16	1/60	50 mm
	picado posición 2	RAW	RGB	200	F16	1/60	50 mm
	contrapicado posición 2	RAW	RGB	200	F16	1/60	50 mm
Pedestal	frontal	RAW	RGB	200	F16	1/13	50 mm
	picado	RAW	RGB	200	F16	1/13	50 mm

Fig. 3.3. Tabla con ajustes de cámara

Captura fotogramétrica

La plataforma rotatoria empleada durante las sesiones fotográficas permitió realizar un total de 48 fotografías por cada giro completo de la pieza, rotando un ángulo de 7.5° entre imágenes consecutivas. La metodología empleada para fotografiar cada parte del modelo varió ligeramente:

-Se decidió no encuadrar la totalidad del cuerpo del Efebo en cada imagen. En su lugar, se optó por acercar la cámara, y hacer que la composición de fotografías a varias alturas, tomadas con el mismo ángulo respecto a la escultura y con una superposición de al menos $\frac{1}{3}$ entre imágenes, diese la totalidad de la altura de la figura. Además, se tomaron fotografías picadas y contrapicadas para obtener información de aquellas zonas superiores e inferiores del modelo que quedaban ocultas en las tomas frontales (fig 3.4). Así se pudo aumentar la calidad de detalle captado.

-Esta medida no se consideró necesaria para las tomas de los fragmentos más pequeños, por lo que las fotografías se realizaron encuadrando las piezas enteras. En el caso del brazo y de la pierna, se realizaron tomas colocando la pieza en varias posiciones (fig. 3.5 y 3.6) sobre una base con soportes metálicos que sujetaban la pieza, evitando así perder información de áreas ocultas. Para escanear el fragmento de pedestal (fig. 3.7) no fue necesario emplear soportes metálicos ni voltear la pieza, ya que no se consideró necesario digitalizar el volumen inferior.

Para la captura del cuerpo se realizó un total de 7 bucles: 3 frontales, 3 picadas y 1 contrapicada. (esquema en fig. 3.4 y fotografías en fig. 3.8, 3.9 y 3.10). Para la captura de la pierna se realizaron 6 tomas, 3 (toma frontal, toma picada y toma contrapicada) con la pieza en una posición, y otras 3 (toma frontal, toma picada y toma contrapicada) con la pieza girada 180° (esquema en fig. 3.5 y fotografías en fig. 3.12). Para la captura del brazo se realizaron 4 bucles, 2 (frontal y picada) con la pieza en una posición, y otras 2 (frontal y picada) con la pieza girada 180° (esquema en fig. 3.6 y fotografías en fig. 3.11). Finalmente para la captura del fragmento de pedestal se realizaron 2 bucles, un en vista frontal y otro en vista picada (esquema en fig. 3.7 y fotografías en fig. 3.13).

A continuación se muestran tanto los esquemas preparatorios de los diferentes procesos de captura como algunas de las fotografías tomadas durante el proceso (fig. 3.4 a 3.13).

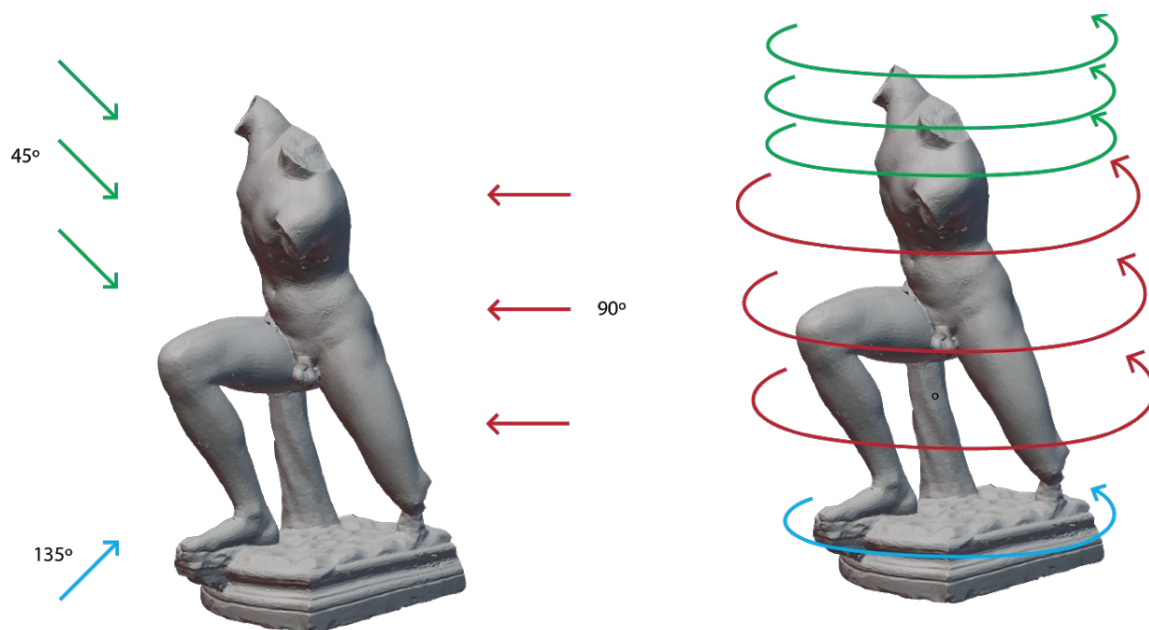


Fig. 3.4. Modelo de tomas fotográficas: cuerpo

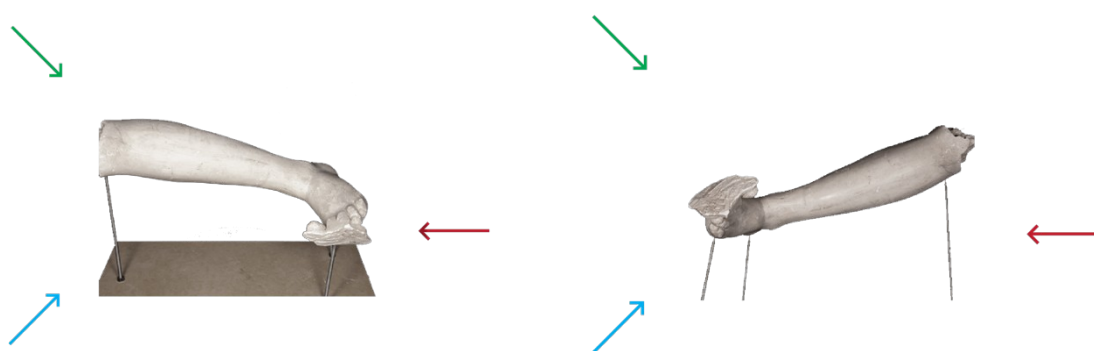


Fig. 3.5. Modelo de tomas fotográficas: pierna

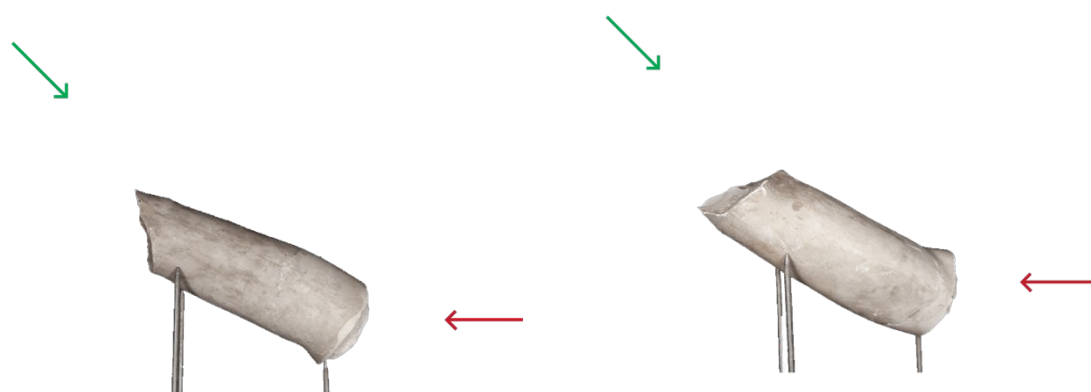


Fig. 3.6. Modelo de tomas fotográficas: brazo



Fig. 3.7. Modelo de tomas fotográficas: pedestal



Fig. 3.8. Tomas fotográficas del cuerpo



Fig. 3.9. Tomas fotográficas picadas del cuerpo



Fig. 3.10. Tomas fotográficas contrapicadas del cuerpo

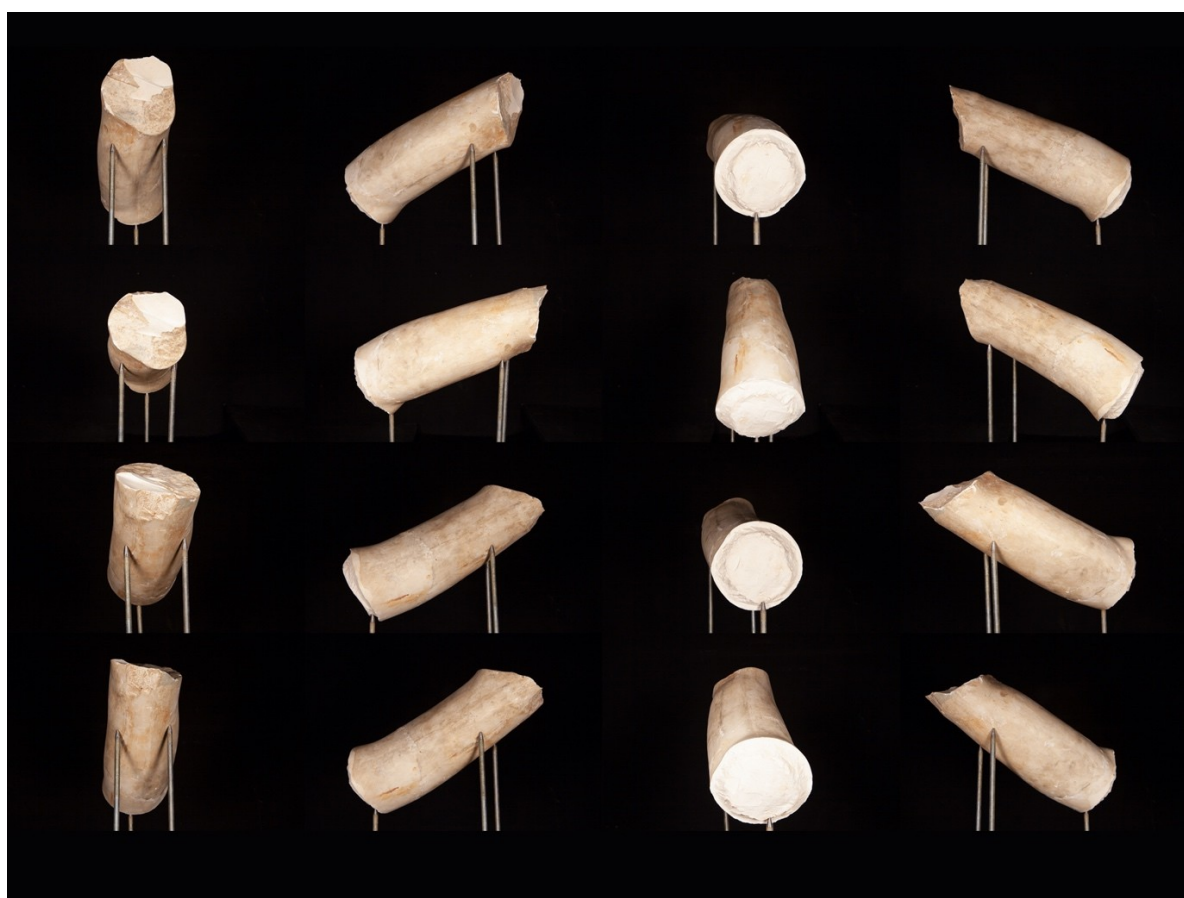


Fig. 3.11. Tomas fotográficas del brazo

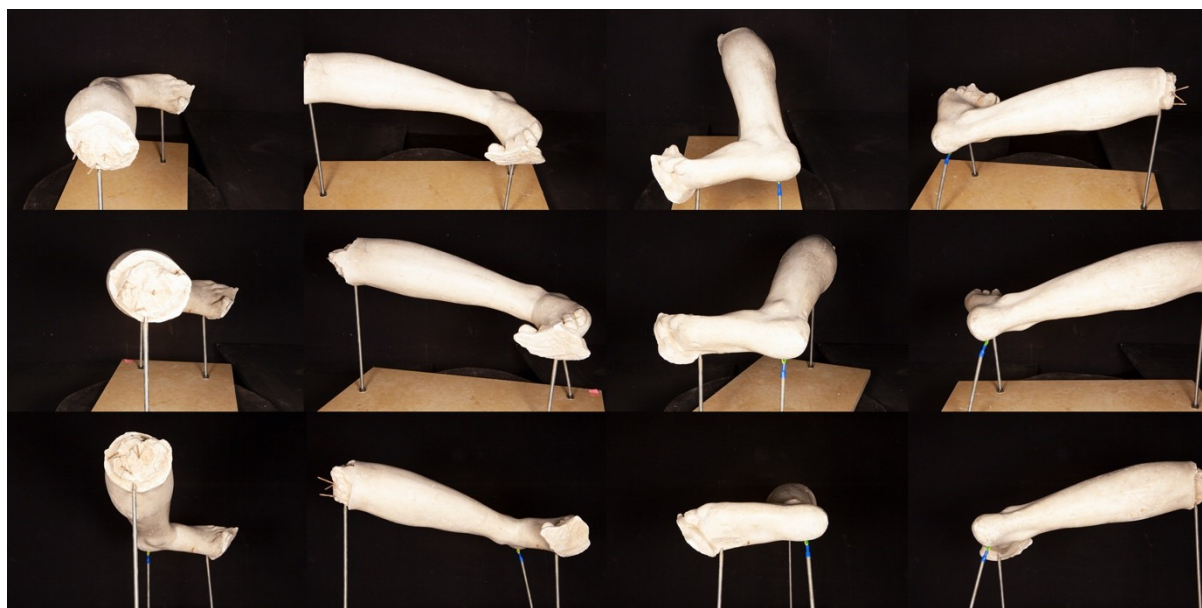


Fig. 3.12. Tomas fotográficas de la pierna



Fig. 3.13. Tomas fotográficas del fragmento de pedestal

4. RESTITUCIÓN FOTOGRAMÉTRICA CON MESHROOM

Santiago Tena Morales

Identificador ORCID: 0000-0002-8629-5078

Si mediante la fotografía proyectamos una escena tridimensional en un plano bidimensional, a través de la fotogrametría tratamos de revertir este proceso reconstruyendo la tridimensionalidad preexistente en la escena a partir de cálculos de triangulación en las imágenes fotográficas. Esta técnica se desarrolló inicialmente en el ámbito de la topografía, donde se utilizaba este sistema para generar mapas terrestres a partir de fotografías aéreas.

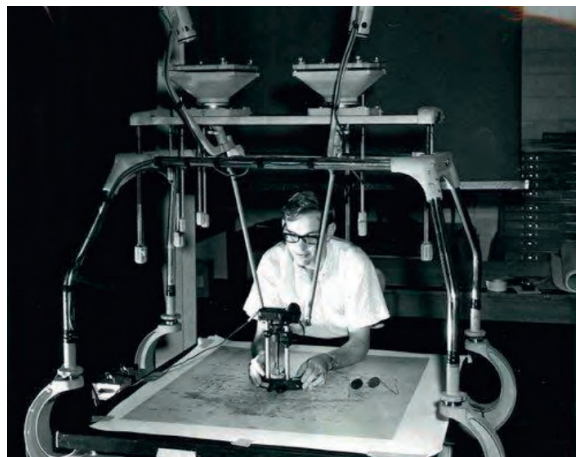
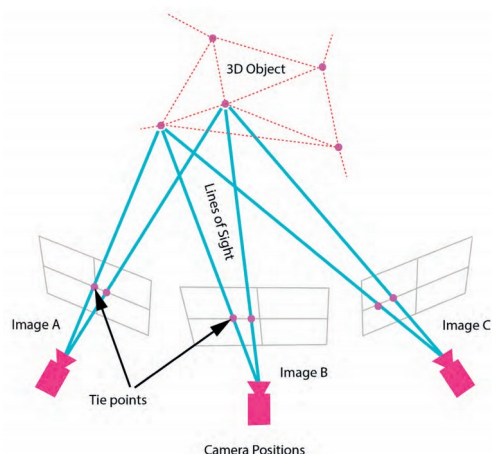


Fig. 4.1. Triangulación para obtener las coordenadas de los puntos en el espacio (izqda). Visor fotogramétrico para usos topográficos (dcha)

Con el desarrollo de los equipos informáticos y un constante aumento de la capacidad de cálculo, se ha hecho posible el procesado de grandes lotes de imágenes, lo que a su vez ha posibilitado que además de mapas topográficos podamos crear reconstrucciones digitales tridimensionales de objetos físicos.

Como ya indicamos en la descripción del proyecto, hemos elegido software open source por la capacidad de acceso al mismo por parte de cualquier usuario de manera libre y gratuita y por su filosofía sobre la difusión del conocimiento. En el caso de la restitución fotogramétrica el software elegido ha sido Meshroom. Se trata de una herramienta desarrollada por la asociación Alicevision, una organización sin ánimo de lucro cuya finalidad es crear una solución informática que democratice el acceso a las tecnologías de digitalización 3D.

Sin ánimo de ahondar aquí en el complejo sistema en el que se basan sus algoritmos para hacer la restitución fotogramétrica, nos limitaremos a explicar los pasos a seguir y las diferentes opciones que nos ofrece este software para conseguir mallas tridimensionales a partir de imágenes tomadas de nuestra escultura de referencia.

Antes de comenzar es importante destacar que las imágenes que hemos tomado de la escultura se han realizado desde un equipo específico, con una configuración de lente y un sensor determinados. Esta configuración aporta a las imágenes una serie de características, como aberraciones y deformaciones, que determinan en gran medida la calidad del proceso de restitución. Dado que actualmente las cámaras digitales son capaces de archivar toda esta información sobre la toma fotográfica en forma de metadatos, estos van a ser importantes en el proceso de cálculo ya que le permiten al software hacer las oportunas correcciones en las distorsiones del conjunto cámara-lente. Meshroom es capaz de extraer esta información de los metadatos de cada imagen, por lo que durante el procesado de las tomas deberemos facilitar la transmisión de estos a la imagen final procesada. Es por ello que la elección del formato de imagen a guardar para nuestras fotografías digitales resulta fundamental. En nuestro caso recomendamos guardar todas las fotografías de la sesión en formato RAW, que es un formato de archivo capaz de almacenar la mayor cantidad de información posible y no presenta pérdidas de calidad.

Organización de la interfaz de Meshroom

Meshroom dispone de seis espacios de trabajo principales desde donde vamos a interactuar con el programa.

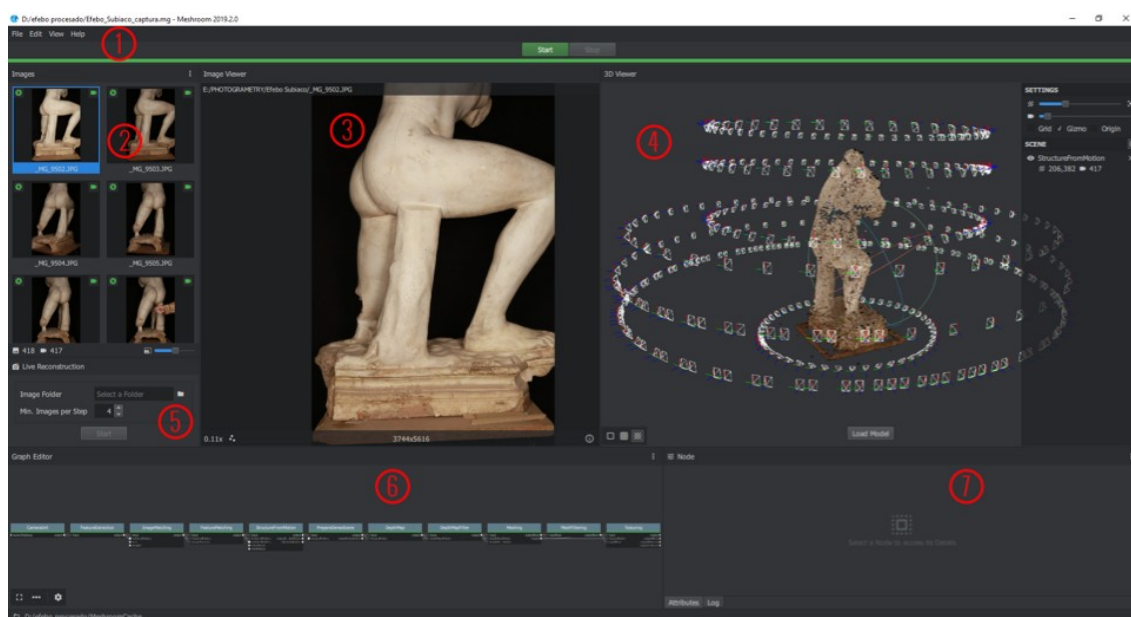


Fig. 4.2. Interfaz de trabajo en Meshroom. 1) Menú superior. 2) Images. 3) Image Viewer. 4) 3D Viewer. 5) Live Reconstruction. 6) Graph Editor. 7) Node

El primer espacio es la barra de **Menú** superior. Se trata de un menú muy sencillo en el que vamos a poder acceder al menú File, con las opciones de creación, guardado y apertura de proyectos y salida del programa; al menú Edit, con los habituales controles para hacer y deshacer pasos; y el menú View, desde donde podremos configurar la visualización de las opciones avanzadas así como del Live Reconstruction.

El segundo espacio es el área denominada **Images**. En este espacio podremos adjuntar las imágenes que vamos a emplear para la reconstrucción. El proceso es rápido e intuitivo, simplemente tenemos que arrastrar y soltar nuestros archivos de imágenes dentro de este área para importar las fotografías en el proyecto.

El tercer espacio se denomina **Image Viewer** y se trata de un visualizador de imágenes. Es útil para observar en detalle cualquier fotografía que seleccionemos para, por ejemplo, localizar imágenes trepidadas o mal enfocadas dentro de un lote. Es importante que las fotografías con problemas de enfoque o movidas sean eliminadas del proceso para evitar errores en el cálculo, ya que la información que ofrecen al software es errónea.

El cuarto espacio es el **3D Viewer**. Aquí podemos ver la estimación del cálculo de las posiciones de las cámaras y cuando el proceso de reconstrucción termine también se mostrará aquí el resultado final de la malla de puntos del objeto una vez restituido su volumen.

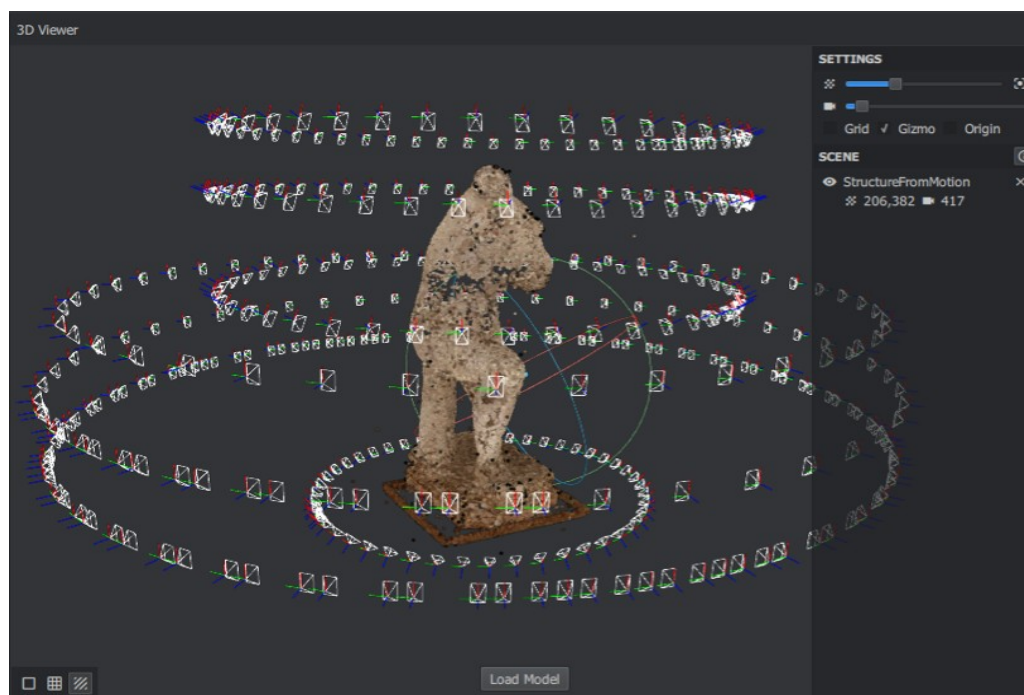


Fig. 4.2. Interfaz de trabajo en Meshroom

El quinto espacio se denomina **Live Reconstruction**. Desde este espacio tenemos un acceso rápido al proceso de restitución. Para iniciar el proceso simplemente indicamos una carpeta donde se encuentren las imágenes a procesar y pulsamos sobre el botón Start (fig. 4.3 izqda).

El sexto espacio es el **Graph Editor**. Desde aquí podremos hacer una configuración avanzada del proceso de reconstrucción. Se trata de un árbol de nodos en el cual cada nodo contiene la configuración establecida por el usuario para definir con precisión cada etapa del proceso. Aunque para una configuración avanzada de estos nodos es necesario amplios conocimientos técnicos, para un usuario medio es suficiente con atender a la configuración de los nodos MeshFiltering y Texturing, en los que si lo deseamos podemos configurar aspectos relativos a la resolución de la malla y a la generación de la textura. En nuestro proyecto la configuración por defecto de todos los nodos resultó óptima para nuestras necesidades.

El séptimo espacio, **Node**, nos ofrece los parámetros de configuración de los nodos que tengamos seleccionados en el Graph Editor.

Proceso de trabajo

Una vez visto cómo se distribuyen los espacios de trabajo, el proceso de reconstrucción es relativamente automático y únicamente requiere un poco de paciencia para la espera de los resultados. El primer paso siempre es la creación de una carpeta dentro de nuestro disco duro que contenga de forma ordenada todos los archivos del proyecto. Para una gestión óptima recomendamos la siguiente organización de la carpeta del proyecto: Crear una carpeta con el nombre de cada pieza a reconstruir y en su interior crear una subcarpeta denominada *Fuentes*, donde guardamos el lote completo de fotografías de dicha pieza, y otra denominada *Salida*, donde Meshroom guardará el conjunto del proyecto y generará una serie de carpetas con los resultados parciales de cada fase del proceso. Y es que aunque la restitución fotogramétrica se va a realizar de forma progresiva y continua siguiendo las instrucciones especificadas en el árbol de nodos, en cada uno de sus subprocesos genera una serie de archivos que se almacenan en un conjunto de carpetas que el programa crea automáticamente. A continuación mostramos este conjunto de carpetas que Meshroom crea dentro de cada proyecto (fig. 4.3 dcha). El fichero final con la malla tridimensional lo podemos encontrar en dos ubicaciones: por un lado en la carpeta *Meshing*, donde almacena el fichero con la reconstrucción de la pieza sin texturizar en formato OBJ, y por otro lado en la carpeta *Texturing* donde, además de la malla en formato OBJ, guarda su textura.

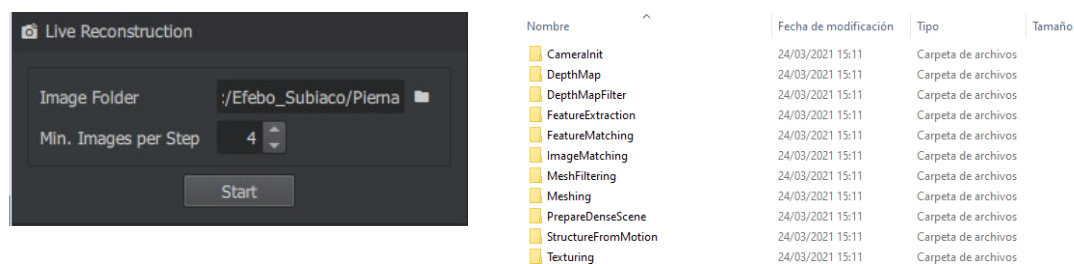


Fig. 4.3. Panel Live Reconstruction (izqda). Estructura de carpetas del proyecto creadas por Meshroom (dcha)

Una vez creado y guardado cada proyecto procedemos a iniciar el proceso desde el panel Live Reconstruction, indicando la ubicación de la carpeta *Fuentes*, e iniciando el proceso con sobre el botón *Start*. Una ventaja de la visualización del árbol de nodos que nos ofrece el Graph Editor es que nos permite conocer el estado global del proceso asignando un código de color al estado de cada nodo, donde señala en verde cada proceso terminado, en naranja el proceso en curso y en azul los procesos pendientes (fig. 4.4).

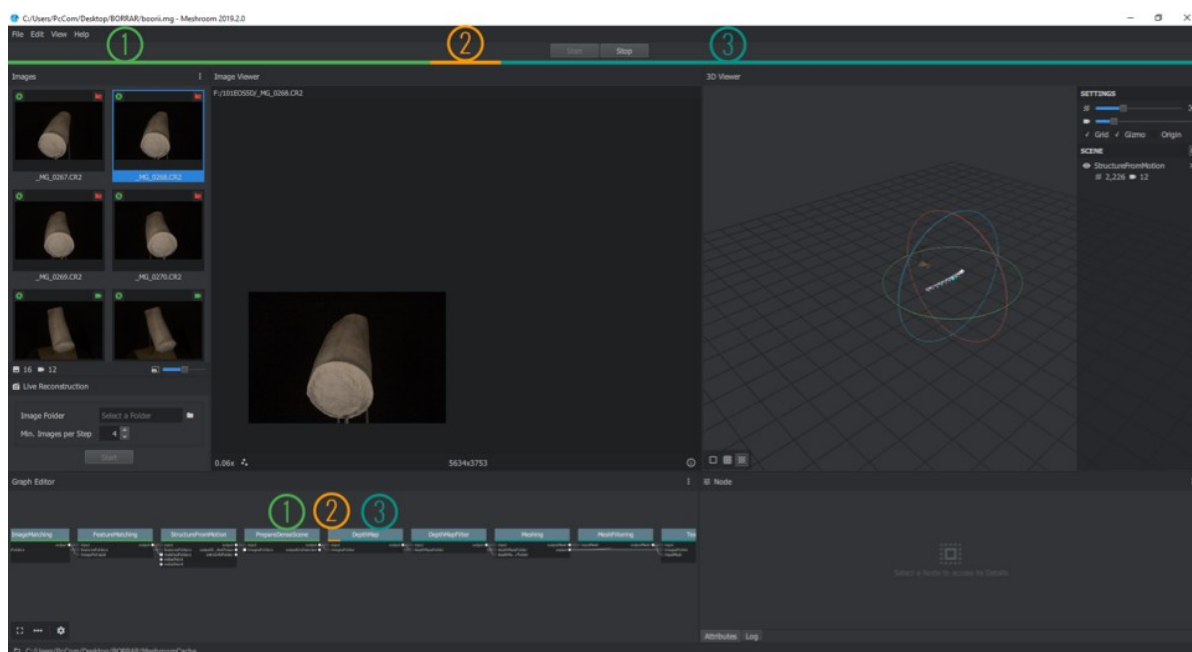


Fig. 4.4. Visualización del estado de cada proceso en el Graph Editor

Transcurrido el tiempo de procesado se obtiene finalmente el objeto tridimensional con y sin textura en las citadas carpetas *Meshing* y *Texturing*.

5. UNIÓN Y REESCALADO DE LAS PIEZAS EN BLENDER

Iria Groba Martín

Identificador ORCID: 0000-0002-2172-0993

Una de las limitaciones del proceso de restitución fotogramétrica es que el software carece de información de escala como para interpretar el tamaño real de los objetos. Debido a que para este proyecto era necesario crear digitalmente unos nuevos fragmentos que, una vez impresos, encajasen con precisión en la escultura real, era imprescindible dar a los modelos digitalizados su tamaño adecuado. Para componer los diferentes fragmentos escaneados y darles su dimensión exacta se utilizó Blender. Como podemos observar en la fig 5.1, se tomaron las medidas de las zonas más anchas de las distintas partes de la escultura y esta información se trasladó a Blender creando prismas rectangulares con las dimensiones obtenidas. Las piezas escaneadas se escalaron empleando estas formas como guía (fig 5.2)



Fig. 5.1. Metodología empleada para medir las diversas partes de la esculturas

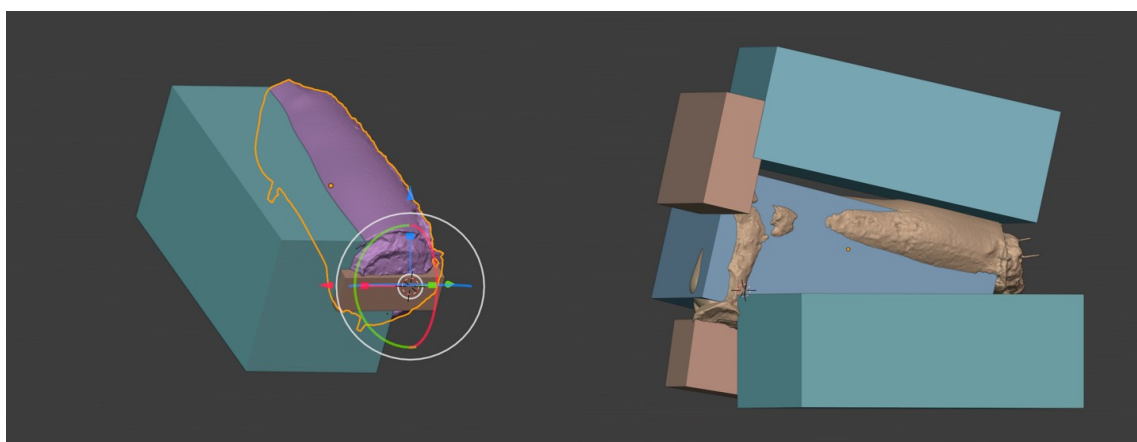


Fig. 5.2. Escalado de los modelos en Blender

Para unir las piezas se estudió el ángulo de inclinación de estas juntando las distintas partes de la escultura y tomando fotografías del conjunto desde varios puntos de vista. Estas imágenes se usaron como referencia para establecer la posición e inclinación de los distintos elementos. La figura 5.3 muestra la comparación entre las fotografías y la posición de los elementos en Blender, y la figura 5.4 presenta el proceso de composición entre los distintos modelos.



Fig. 5.3. Comparativa entre la inclinación del brazo en la escultura original (izqda) y en el modelo tridimensional (dcha)

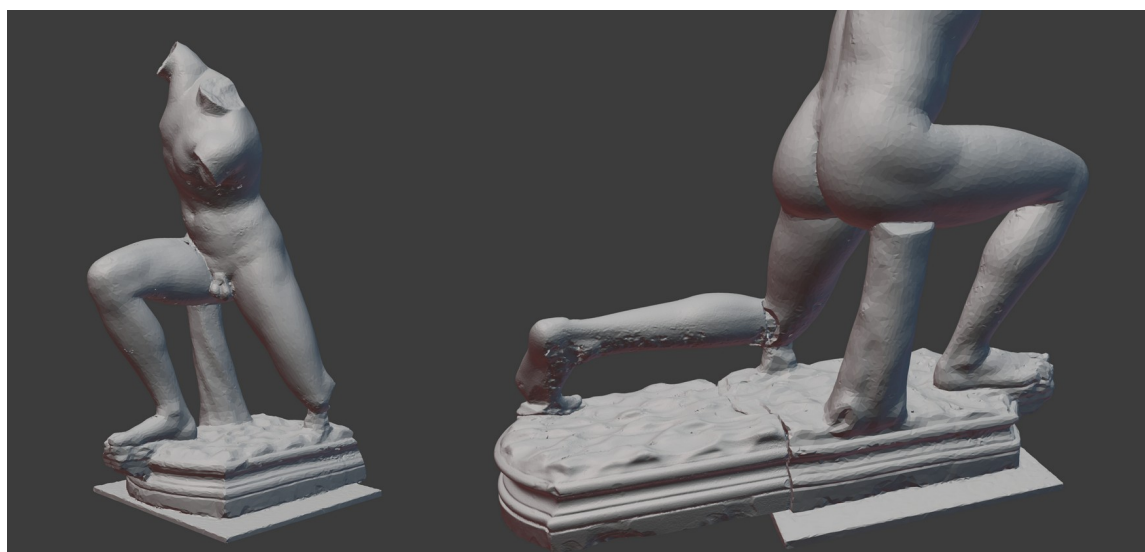


Fig. 5.4. Proceso de unión de las piezas en Blender

Una vez escalado cada fragmento escaneado a su tamaño real y unidos digitalmente todos ellos obtenemos el modelo de la escultura del Efebo (fig. 5.5) que servirá como punto de partida para la construcción de los elementos faltantes, cabeza y brazos.



Fig. 5.5. Resultado final de la composición de los distintos fragmentos del Efebo escaneados independientemente

6. CREACIÓN DE LOS BRAZOS CON BLENDER

Miguel Pozas Pérez

Identificador ORCID: 0000-0003-4952-1256

Planteamiento de la reconstrucción. Período y estilo

Pese a que la autoría y datación del Efebo de Subiaco no está todavía consensuada, ciertas cualidades aspectuales de la obra, tales como la suavidad de la musculatura, el movimiento del cuerpo en varias direcciones o el decidido dominio del espacio tridimensional de los miembros, remiten a las características del período post-clásico que encontramos en obras de autores como Praxíteles o Scopas. Resulta por tanto imprescindible que nuestra recreación de los elementos faltantes comparta esa misma suavidad en las transiciones de la musculatura y esa anatomía poco marcada característica de dicho período post-clásico.

Tampoco hay consenso en lo relacionado con la postura y actitud del personaje representado. Una de las opiniones más aceptadas es la del profesor e historiador Rhys Carpenter¹, quien lo define como: "la convención arcaica de la Knielauf... adoptada en su valor nominal literal: el joven no está corriendo, pero está en la posición arcaica del corredor" (Carpenter 1941: 25).

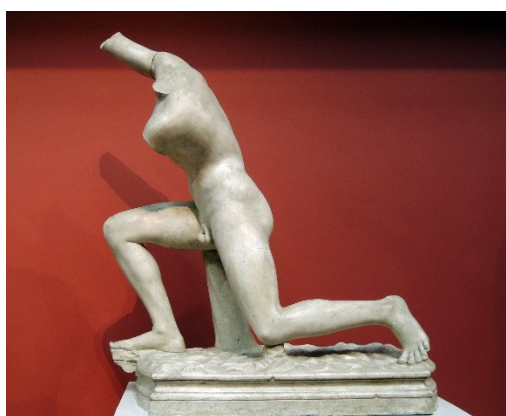


Fig 6.1. Efebo de Subiaco

Es una opinión un tanto problemática ya que resulta ciertamente complicado identificar dicha posición de carrera en esta figura a tenor del escorzo del torso, el brazo derecho proyectado tan vertical y la rodilla prácticamente en contacto con el suelo, y no parece que exista mayor documentación sobre aquello a lo que Carpenter se refería cuando habla de "la posición arcaica del corredor". No obstante, y después de realizar algunas pruebas con el modelo 3D fue finalmente posible imaginar una colocación de los brazos que insinuaría una posición de carrera.

Análisis del escaneado y su postura

Para el trabajo de escultura digital un software de modelado 3D libre como Blender resulta idóneo en tanto que permite una fácil interacción y visualización para la construcción tridimensional de la cabeza y los brazos faltantes. El modelo 3D tal y como se obtuvo tras la restitución fotogramétrica presentaba una gran densidad de polígonos, haciendo muy lento el conjunto de tareas de modelado. De cara a poder trabajar con la suficiente fluidez resulta recomendable mantener el número de polígonos por debajo del umbral de 1 millón, pero nuestro modelo tenía una geometría mucho más densa, lo que nos obligó a aplicar el modificador *Decimate* para reducir el número de polígonos en aquellas partes menos relevantes al tiempo que mantuvimos la densidad original en las áreas que lindan directamente con los brazos y el cuello, pues la precisión en dichas superficies de fractura va a resultar esencial para asegurar un encaje exacto entre los fragmentos nuevos y la escultura real.

¹ Rhys Carpenter, "Observations on Familiar Statuary in Rome", *Memoirs of the American Academy in Rome* 18, Observations on Familiar Statuary in Rome (1941).

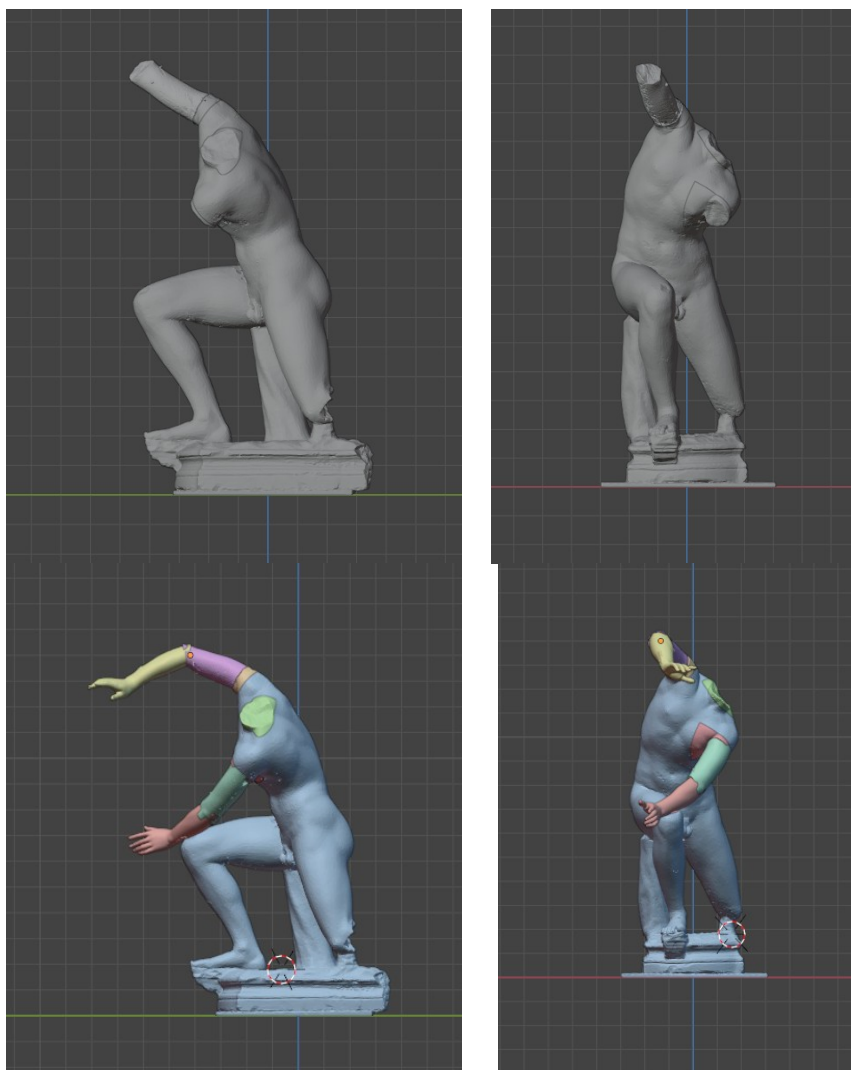


Fig. 6.2. Figura escaneada (superior) y bocetos preparatorios para la posición de los brazos (interior)

Antes de encajar la pose es imprescindible analizar la información que nos ofrece la anatomía de la escultura para poder plantear adecuadamente los brazos en una configuración plausible. Un primer examen nos permite observar no solo que la inclinación del torso y la dirección del hombro izquierdo proyectan el brazo hacia delante, sino también nos revela la existencia una pequeña superficie de desgaste en la rodilla derecha que indica un punto de apoyo del antebrazo izquierdo. Esto significa que la posición del brazo izquierdo era inevitablemente estirada hacia adelante.



Fig. 6.3. Posición del hombro izquierdo y superficie de desgaste en la rodilla derecha

En el caso del brazo derecho solo falta el antebrazo y la mano. En la superficie fracturada del codo se puede observar un pequeño valle que indica una ligera flexión de dicho antebrazo. Empleando para esta fase de boceto fragmentos de brazos de otras esculturas a modo de guía fue posible empezar el proceso de esculpido 3D de los brazos con una idea mucho más clara.

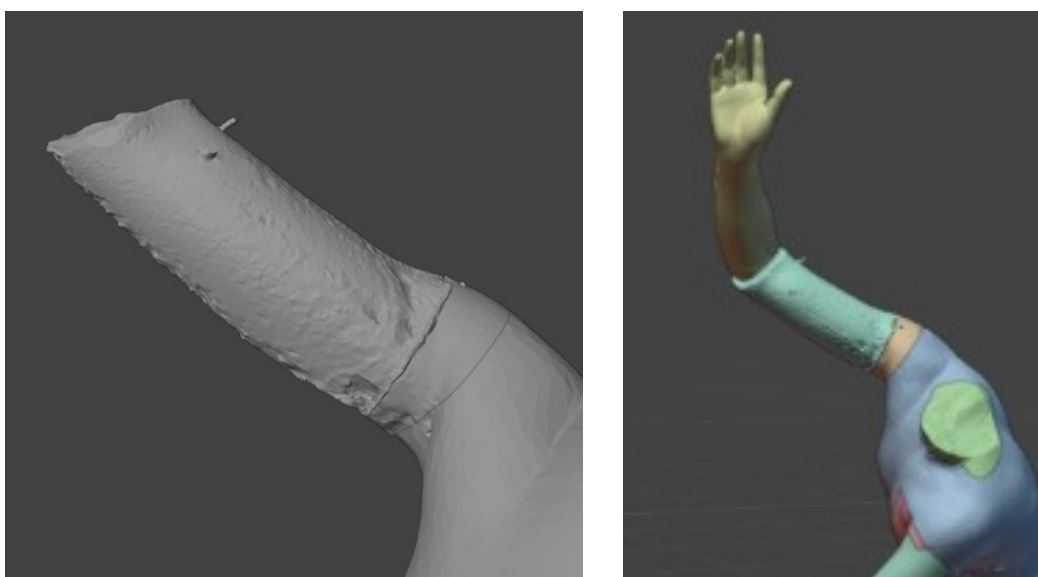


Fig. 6.4. Brazo derecho (izqda) y boceto para la posición del antebrazo faltante (dcha)

Escultura digital de los brazos

Las herramientas que hemos utilizado para la escultura digital o *sculpting* de los brazos han sido esencialmente los siguientes:

Clay brush: Pincel que permite añadir o retirar volumen. En modelos complejos se recomienda el uso de este pincel con la opción *Dyntopo* activada, de modo que se generan nuevos polígonos a medida que vamos trabajando, evitándose así estirar en exceso la malla y manteniendo siempre una topología homogénea.

Smooth brush: Este pincel suaviza el relieve de la geometría, algo que resulta enormemente útil tanto para el acabado final como para controlar los volúmenes a medida que vamos trabajando. Pincel imprescindible en prácticamente cualquier proyecto, en este caso lo ha sido todavía más, ya que aquí buscábamos construir una musculatura particularmente poco marcada y con transiciones suaves.

Crease brush: Este pincel resulta útil a la hora de producir concavidades y convexidades. En su modo de trabajo principal provoca una concavidad en la superficie a la que se aplique, cómo si hincáramos una uña o un palillo de modelar, mientras que en su modo inverso el resultado es como si pellizcáramos la malla. Herramienta imprescindible para detalles como los que encontramos en las uñas.

Grab brush: De uso más ocasional, resulta útil cuando es necesario mover grandes áreas de geometría para ajustar su posición.

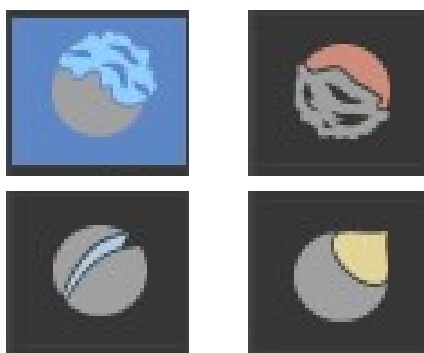


Fig. 6.5. Principales pinceles de escultura. *Clay brush* (superior izqda), *Smooth brush* (superior dcha), *Crease brush* (inferior izqda) y *Grab brush* (inferior dcha)

Existen múltiples formas de empezar un esculpido digital. En este caso decidimos comenzar a partir de simples cilindros independientes para los volúmenes del brazo, antebrazo, palma y dedos, lo que nos permite trabajar individualmente cada uno de ellos y mantener total libertad para reposicionarlos. A medida que íbamos aportando detalle con los pinceles anteriormente descritos los fuimos uniendo mediante el modificador *Boolean* para conseguir una malla unificada y una transición limpia entre los miembros.

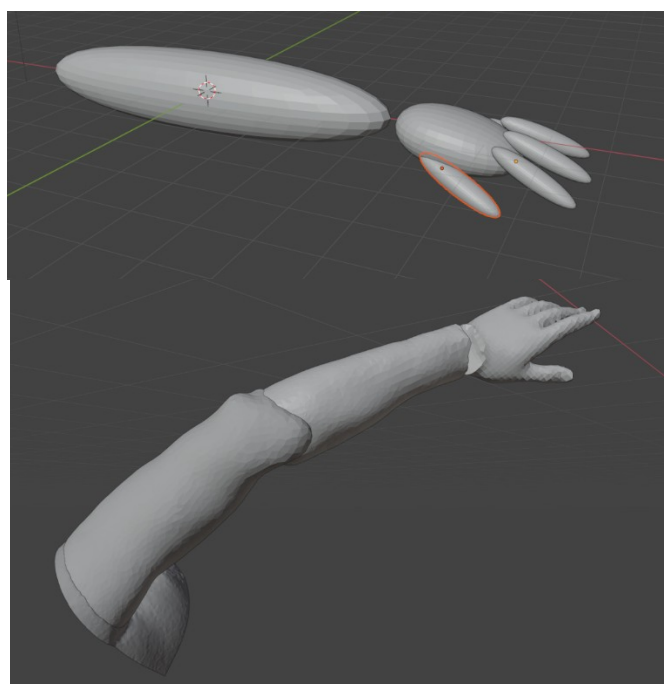


Fig. 6.9. Geometría base para el brazo, antebrazo y mano derecha a partir de cilindros (superior). Esculpido en proceso (inferior)

Una vez finalizada la parte superior del brazo izquierdo, abordar su antebrazo fue mucho más sencillo. Y una estrategia para ahorrar trabajo fue utilizar el antebrazo izquierdo recién construido como base para proyectar el derecho, al que solo fue necesario realizar algunos cambios menores para adaptarlo a su postura.

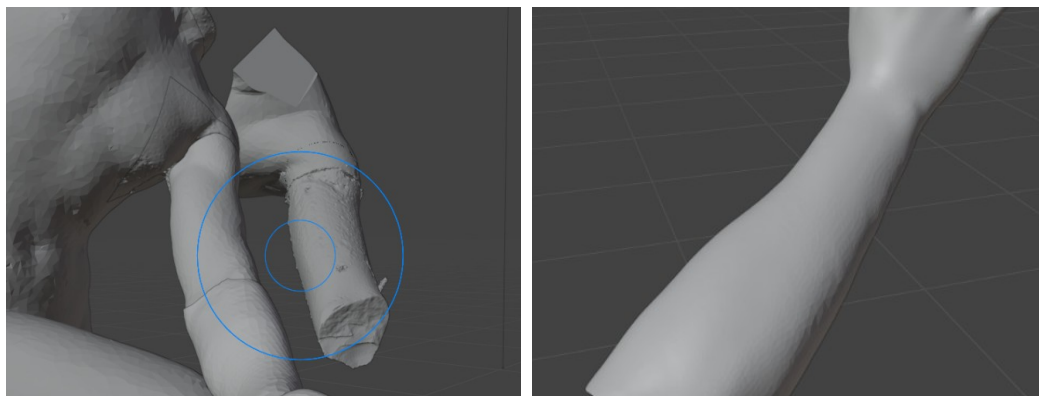


Fig. 6.10. Brazo derecho escaneado como referencia para el brazo izquierdo (izqda). Antebrazo derecho terminado (dcha)

En el caso de los dedos el procedimiento fue muy similar: se procedió modelando en detalle un índice y un pulgar y posteriormente se copiaron y pegaron para generar el resto de dedos. Una vez colocados en su posición definitiva se utilizó aquí de nuevo el modificador *Boolean* para conseguir una malla unificada e integrar cada dedo con el conjunto de la mano.

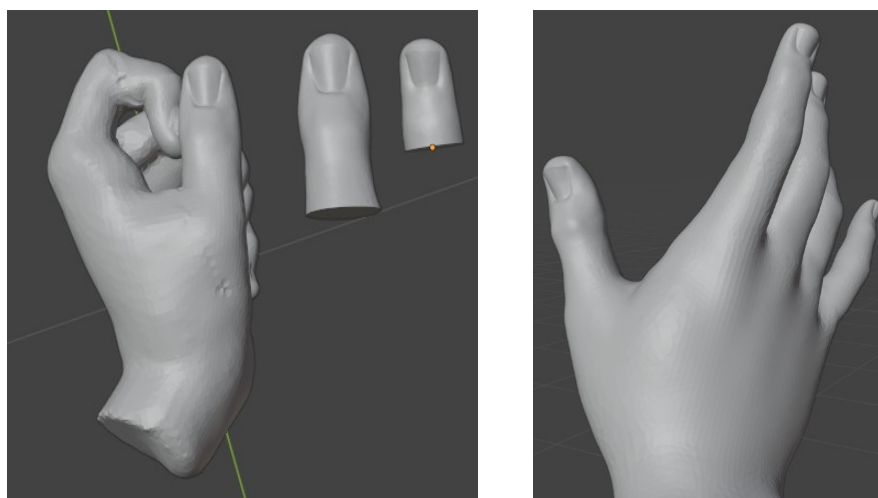


Fig. 6.12. Mano izquierda, dedos pulgar e índice modelados individualmente (izqda) y mano derecha en proceso (dcha)

Como con la información que nos aporta la anatomía de la escultura resulta imposible obtener ninguna certeza sobre la posición de las manos, ésta se ha determinado atendiendo únicamente a factores compositivos buscando reforzar la citada posición de carrera que Carpenter propuso para esta figura.

Una vez los brazos están finalizados solo restaría provocar su intersección con el resto de la escultura empleando para ello de nuevo el modificador *Boolean*, consiguiendo así una superficie de fractura exacta que permita que una vez impresas estas piezas encajen de forma precisa con la escultura original.



Fig. 6.13. Plano de detalle de la intersección entre antebrazo y brazo derecho

Una vez las nuevas piezas fueron cortadas empleando la figura original como volumen de fractura, el resultado final es el siguiente:

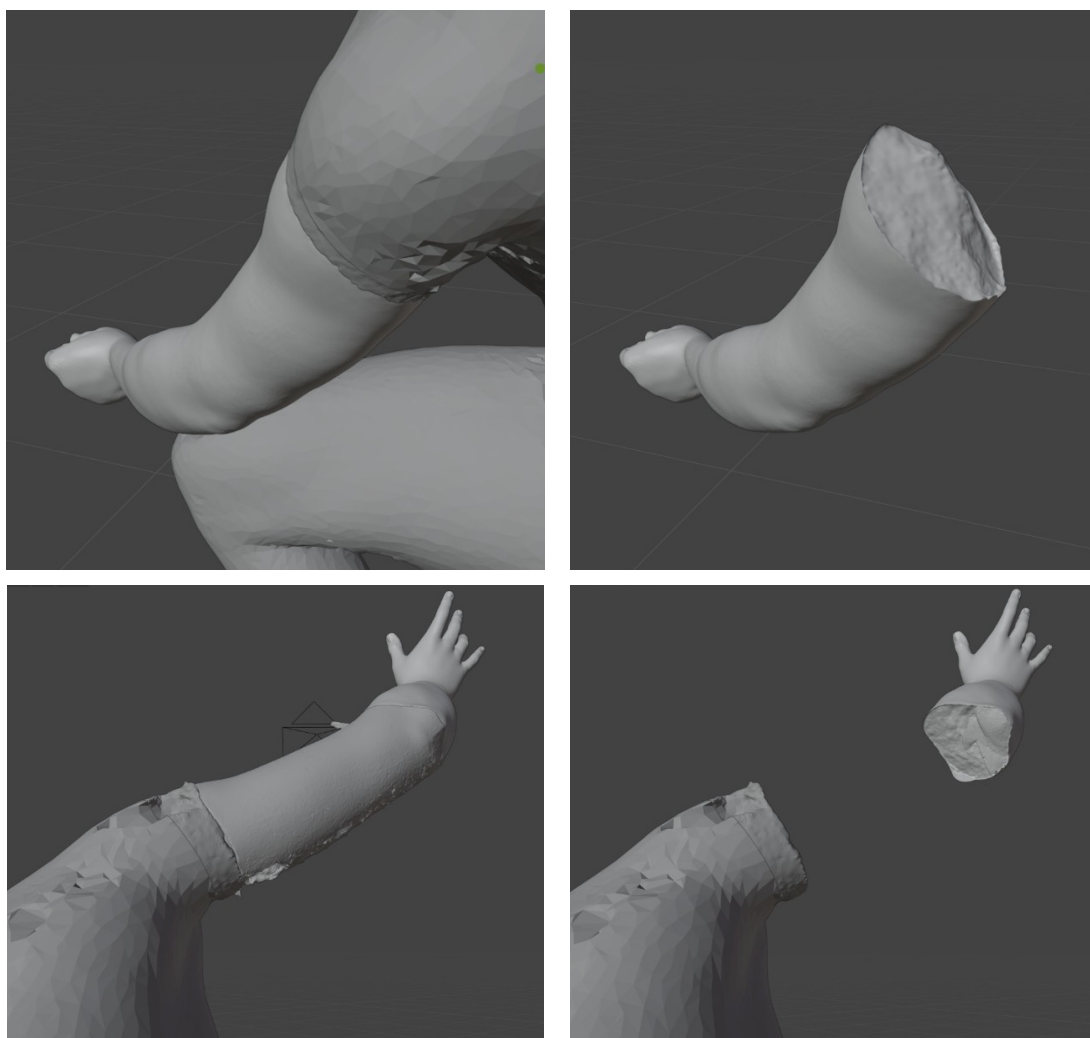


Fig. 6.14. Precisas superficies de fractura generadas mediante el modificador *Boolean*

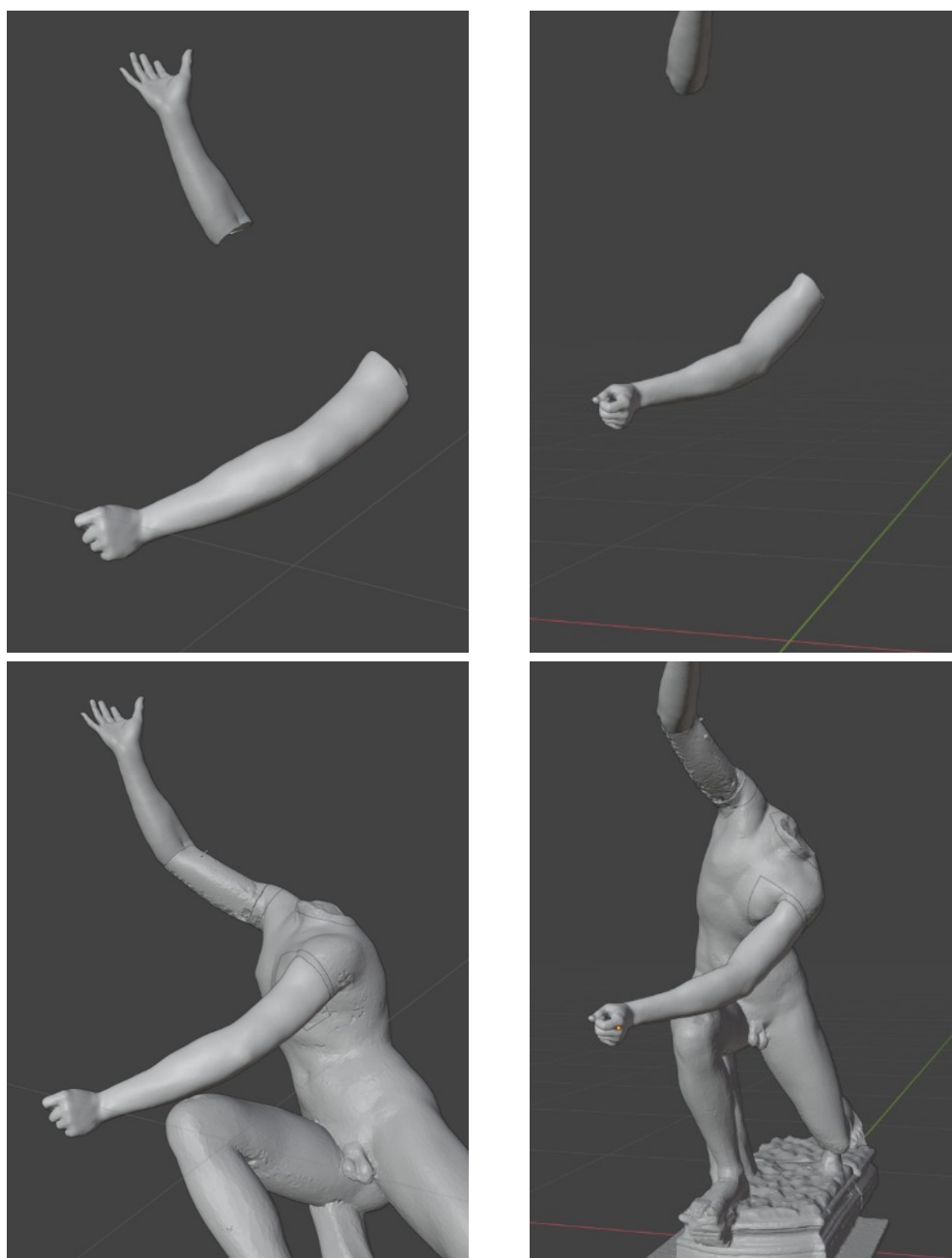


Fig. 6.14. Resultado final

7. MODELADO 3D Y ESCULTURA DIGITAL DE LA CABEZA CON BLENDER

Darío Lanza Vidal

Identificador ORCID: 0000-0003-0475-2114

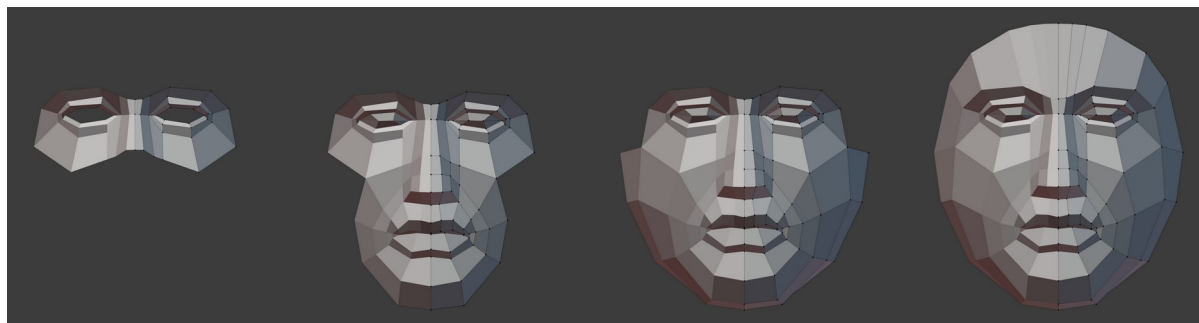
Dado que la cabeza original se ha perdido, hemos de construir una nueva tratando de mantener la máxima coherencia estilística con la obra en su conjunto, para lo cual nos deberemos apoyar en el estudio de piezas del período post-clásico al que se atribuye esta obra, como el Apolo de Belvedere, el Apollino Medici y el Hypnos del Museo del Prado (fig. 7.1). En una observación en detalle de las cabezas de estas obras podemos extraer ciertas características estilísticas comunes, como la suavidad de los rasgos del rostro, el gesto menos severo y más juvenil que en el período clásico y la riqueza del trabajo en la cabellera, con un elaborado desarrollo de rizos, mechones y moños, elementos que aparecen en este período como una constante y que por tanto nos interesará incluir en nuestra obra.



Fig. 7.1. Apolo de Belvedere (izquierda), Hypnos del Museo del Prado (centro) y Apollino Medici de la Gallerie degli Uffizi (derecha)

Modelado poligonal del rostro

Para la construcción de los elementos faltantes hemos recurrido a Blender, programa de creación 3D libre y que, por tanto, se halla disponible de forma gratuita. La creación de la cabeza comienza con el modelado poligonal a partir de un plano rectangular, al cual activamos el trabajo en simetría y comenzamos a evolucionar desarrollándolo mediante la extrusión y prolongación de sus vértices para ir cubriendo toda la superficie del rostro, manteniendo en todo momento como referencia los referentes que hemos mencionado (fig. 7.2). En esta fase ha resultado esencial mantener un adecuado flujo de polígonos que permita representar adecuadamente las facciones y la anatomía del rostro al tiempo que mantengamos el número de polígonos en el mínimo posible para facilitar la edición del modelo².



² De cara a proyectar una correcta topología para el modelado poligonal del rostro recomendamos el tutorial *A Better Face Tutorial*, disponible en <https://www.blender-models.com/articles-tutorials/modeling/a-better-face-tutorial/>

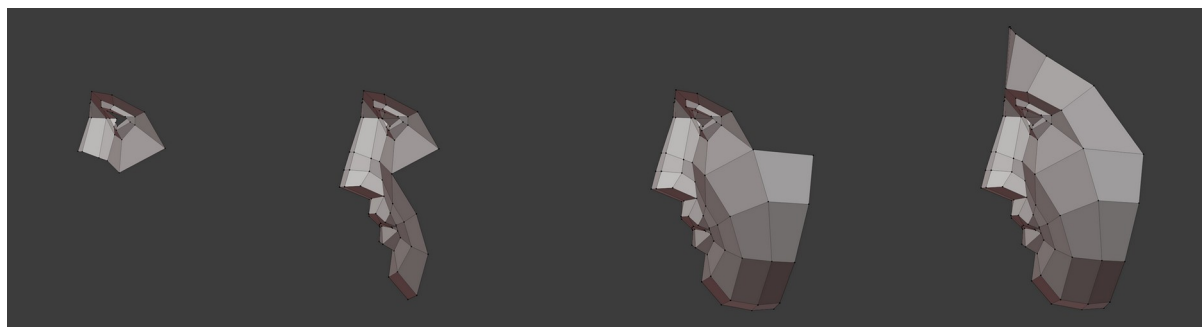


Fig. 7.2. Modelado poligonal del rostro a partir de un plano

La oreja, debido a su particular geometría, fue modelada separadamente y fusionada con el resto de la cabeza en un espacio reservado al efecto. Es especialmente importante para este paso construir la oreja empleando el mismo número de polígonos que el hueco dejado en la cabeza, para que el flujo de polígonos encuentre continuidad entre ambas piezas (fig. 7.3).

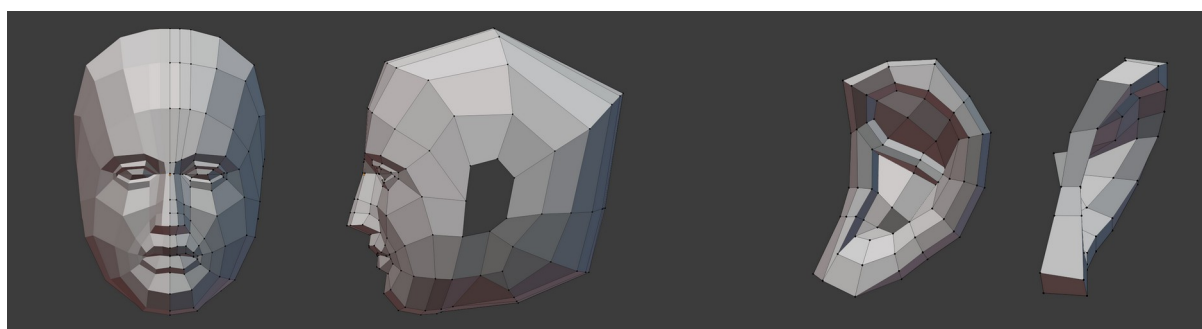


Fig. 7.3. Modelado de la oreja como elemento independiente

El cuello se ha obtenido a partir de la extrusión de los polígonos situados en la base de la cabeza. Llegados a este punto, una vez fusionada la cabeza con la oreja y extruido el cuello, podemos proceder a subdividir la geometría, empleando en nuestro caso el modificador *Subdivision Surface*. Un sólo nivel de subdivisión es suficiente en este momento para disponer de una visión suavizada y curvilínea de la anatomía de la cabeza. La cabellera, en la cual habrá que realizar un pormenorizado trabajo escultórico, la modelamos como una geometría independiente, a partir de un duplicado de la parte superior del cráneo. Dado que vamos a poner una cinta en el pelo semejante a la que presentan las esculturas que nos sirven de referencia, creamos esta a partir de una curva *Path* a la que damos grosor empleando otra curva circular que podemos modificar para obtener la sección deseada (*Object Bevel*). Ambos elementos, cabellera y cinta, son subdivididos y serán fusionados con el volumen principal de la cabeza en una fase posterior (fig. 7.4). Trabajar estos elementos de forma independiente nos proporciona una mayor capacidad de edición y un trabajo escultórico más seguro, al evitar que el esculpido en un objeto afecte accidentalmente a otro.

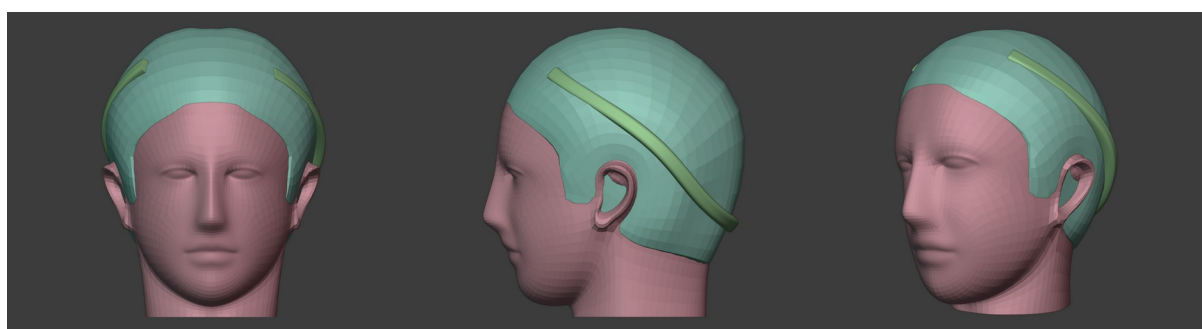


Fig. 7.4. Cabeza, cabellera y cinta como elementos independientes que serán posteriormente fusionados en una sola pieza

Escultura digital de la cabeza y cabellera

Una vez que la estructura poligonal básica está terminada podemos comenzar a la fase de esculpido digital o *digital sculpting*, mediante la que modificaremos la forma del modelo de una manera flexible e intuitiva, como si estuviera realizado en arcilla. Antes de comenzar deberemos aplicar los modificadores *Mirror* (de simetría) y *Subdivision Surface* (de subdivisión) para convertir toda la geometría virtual en geometría real, y activaremos el modificador *Multiresolution*, que proporciona un sistema de subdivisión específicamente diseñado para el esculpido con el que podremos movernos por los diferentes niveles de subdivisión para realizar las labores de esculpido según el nivel de detalle que necesitemos en cada momento.

Para dar forma a la cabellera han resultado especialmente útiles los pinceles *Inflate* y *Blob*, con los que conseguimos dar volumen a los mechones y rizos, *Grab* para desplazarlos lateralmente, *Crease* para agudizar las aristas e incisiones y *Smooth* para suavizar las superficies (fig. 7.5 izqda). Para conseguir un resultado natural es imprescindible diseñar previamente el trazado de cada mechón, buscando un recorrido natural y dinámico a cada uno de los rizos, siguiendo su movimiento y evoluciones a lo largo de la cabeza de modo que cada mechón sea único y presente un desarrollo individual, entremezclándose con los demás y evitando un trazado aleatorio o una formulación excesivamente mecánica que restaría calidad a nuestra escultura (fig. 7.5 dcha).

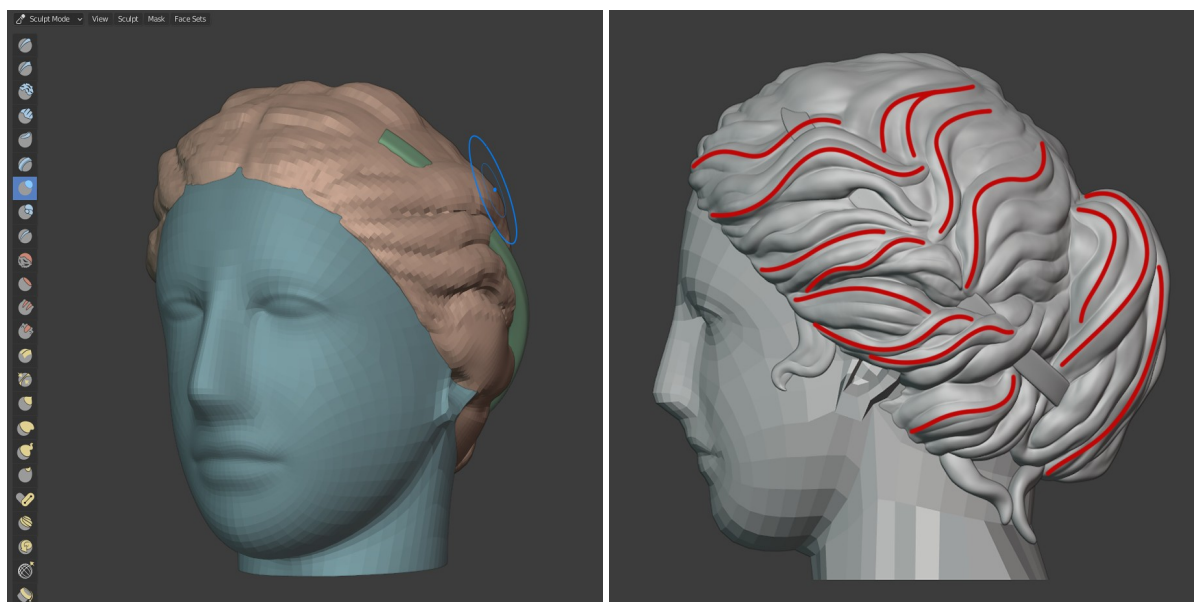


Fig. 7.5. Pinceles de escultura para dar volumen a la cabellera (izqda). Diseño de un trazado individual para cada rizo y mechón (dcha)

De igual modo se procede con el resto de la cabeza, subdividiendo el objeto con *Multiresolution* y utilizando los citados pinceles de sculpting para dar forma a la cara. Aquí resulta de especial importancia mantener una cierta semejanza con los rasgos de las esculturas de este periodo que tomamos como referencia pero sin reproducir ninguna en concreto, pues la escultura original que tratamos de reconstruir y cuya cabeza se ha perdido habría de tener un rostro que, si bien presentaría ciertas características estilísticas comunes, fue sin duda individual y a su manera diferente a los demás, y por tanto debemos evitar clonar las facciones de una escultura existente sino proponer un rostro distinto, plausible con dicho periodo artístico.

Con la incorporación del moño en la parte superior de la cabeza, las patillas y unos rizos libres en la nuca, todos ellos por el momento como objetos individuales, podríamos dar por finalizada la construcción de la cabeza (fig. 7. 6).



Fig. 7.6. Esculpido del rostro

Si bien hasta este punto ha resultado de gran utilidad construir y desarrollar cada elemento de la cabeza como objetos independientes para protegerlos durante las tareas de esculpido, el proceso de impresión 3D exige que todos ellos conformen una malla única, por lo que debemos ahora proceder a fusionar todas las piezas individuales en una. Para ello es imprescindible aplicar en cada objeto todos los modificadores, de modo que no exista geometría virtual, y fusionarlos mediante operaciones booleanas (empleando el modificador Boolean en modo *Union*). También resulta recomendable haber subdividido cada pieza de modo que el tamaño de los polígonos sea semejante en todas ellas, de modo que tras la fusión obtengamos una malla única con una densidad geométrica homogénea.

Conexión con el cuello y preparación para imprimir

Para que las anatomías de la cabeza y la escultura real muestren la necesaria continuidad debemos adecuar la musculatura del cuello de nuestra cabeza a la preexistente. Para ello es necesario atender a la distribución de los músculos de dicha área, en especial al trapecio y al esternocleidomastoideo, así como a las tensiones que deberían presentar a causa del movimiento y la postura. De nuevo pinceles de sculpting como *Inflate* y *Smooth* resultaron esenciales para dar continuidad a dichos grupos musculares (fig. 7.7).

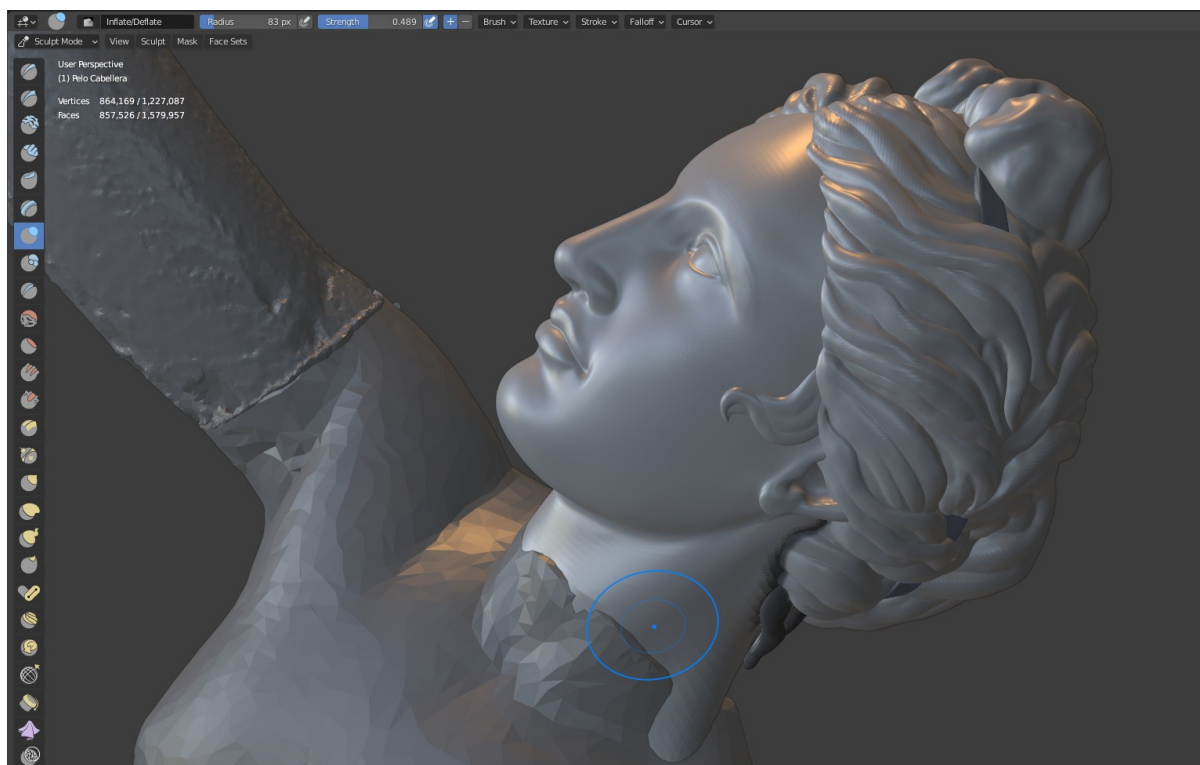


Fig. 7.7. Continuidad anatómica en el cuello

Dado que nuestro modelo debe encajar con precisión con la escultura real existente y que dicha conexión se produce a la altura del cuello, es imprescindible asegurar que la coincidencia entre ambas piezas sea exacta. Para ello, al disponer del modelo 3D del cuerpo podemos proceder a provocar una fractura en la base del cuello de nuestra cabeza logrando que dicha superficie de corte encaje exactamente con la de la escultura. Para esta tarea empleamos una operación booleana (mediante el modificador Boolean en modo *Difference*), obteniendo una superficie con idénticas irregularidades que el área fracturada en la escultura real (fig. 7.8).



Fig. 7.8. Boolean en el cuello para obtener una superficie de fractura precisa

Una vez que tenemos la cabeza correctamente esculpida, asegurada la continuidad de la anatomía y fragmentada por una superficie de corte precisa que garantiza una sutura correcta con la escultura real, llega el momento de materializarla mediante impresión 3D.

8. IMPRESIÓN 3D

Pablo de Arriba del Amo
Identificador ORCID: 0000-0002-8903-8472

El apartado relativo a la materialización de los resultados obtenidos en el proceso de digitalización, reconstrucción y recreación de los fragmentos perdidos de la pieza objeto de estudio habrá de concretarse en un sistema de software y hardware de impresión 3D de carácter libre, tal y como indica la premisa sobre la que se asienta este proyecto.

Dentro de las diversas técnicas de impresión 3D, tanto en software como en hardware libre, ha sido la tecnología de modelado por deposición fundida, o FDM, en la que se ha logrado un mayor desarrollo e implantación. Esta tecnología se basa, esencialmente, en tres elementos: una bobina de filamento de un material que puede ser fundido por temperatura, un extrusor o cabeza de extrusión que calienta el filamento hasta provocar su fusión, y una cama de impresión sobre la que se deposita el material fundido para posteriormente enfriarse y solidificarse. A partir de este planteamiento se ha desarrollado un amplio catálogo de modelos de impresoras con características muy diversas y una amplia oferta de software de apoyo para las mismas.

Uno de los proyectos más representativos en este ámbito de desarrollo en software y hardware libre para impresión 3D es el Proyecto RepRap, que nace con el objetivo de crear una máquina autorreplicable que pueda ser usada para prototipado rápido³ y manufactura. La autorreplicación se concibe como la capacidad para elaborar los componentes necesarios para que la impresora pueda construir una copia de sí misma. La comunidad que sostiene y colabora con el proyecto RepRap ha demostrado una notable capacidad de crecimiento, adaptación y evolución, convirtiéndose en un proceso tecnológico que produce una ruptura en el desarrollo de muy diversos ámbitos, propiciando una renovación radical y logrando afianzar el bajo costo en los procesos de fabricación, en especial en los campos del diseño, la creación y el desarrollo de productos. Se persigue con este proyecto ofrecer a cualquier persona la posibilidad de fabricar objetos para su vida cotidiana.

Dentro del amplio catálogo de impresoras 3D de tecnología por deposición fundida respaldadas por la comunidad de software y hardware de libre acceso existen principalmente dos categorías: el modelo cartesiano y las denominadas impresoras delta.

Las **impresoras cartesianas** reciben este nombre por los desplazamientos en los tres ejes perpendiculares X, Y y Z, siendo el eje de movimiento de la cama de impresión el criterio que se usa para distinguir las dos principales tipologías que encontramos aquí. Así, podemos ver impresoras en las que el cabezal extrusor se desplaza horizontalmente en los ejes X e Y mientras que la plataforma de impresión se desplaza verticalmente en el eje Z, como en las impresoras Makerbot⁴ (fig. 8.1 izqda). Mientras que por otro lado encontramos modelos en los que el cabezal extrusor se desplaza según los ejes X y Z mientras que el movimiento de la superficie de impresión es horizontal hacia adelante y hacia atrás en el eje Y, como el caso de las impresoras tipo Prusa, probablemente la tipología de impresora 3D que mayor acogida ha tenido en la comunidad maker y la cultura DIY (fig. 8.1 dcha).

³ Una máquina de prototipado rápido es una impresora 3D capaz de fabricar objetos en tres dimensiones sobre la base de un modelo hecho en ordenador.

⁴ Empresa que contribuyó al progreso temprano del Proyecto RepRap y cuyos productos son liberados al mercado como tecnologías libres tanto en software (GPL) como en hardware (Open Hardware), lo que permite a los compradores y a otras empresas crear sus propias versiones o modificar los productos que compraron.

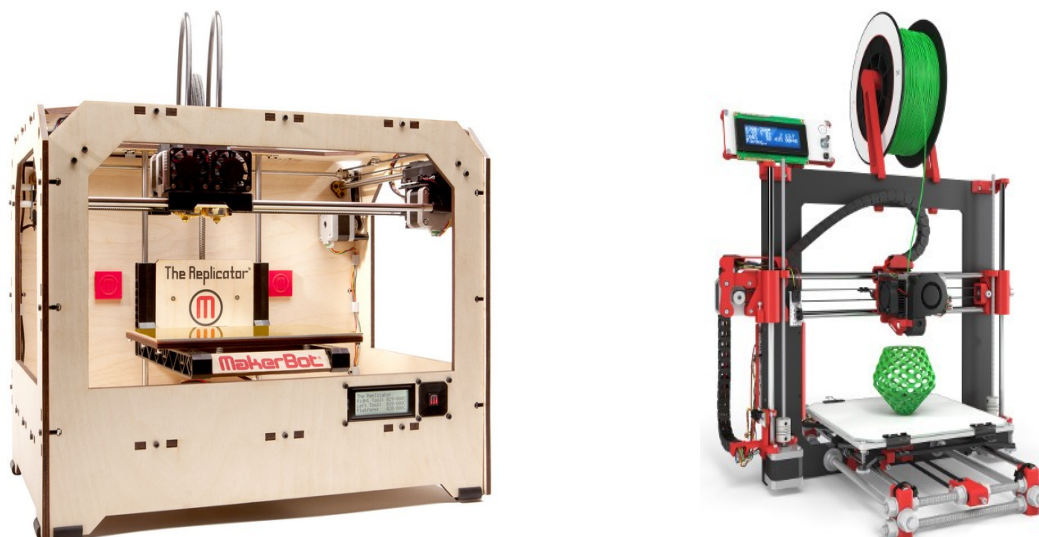


Fig. 8.1. Modelo Replicator de Makerbot (izqda). Impresora cartesiana Prusa i3 (dcha)

Las **impresoras de diseño delta** gozan también de una amplia difusión en la comunidad maker. Se caracterizan por que su cabezal de impresión se haya suspendido en una disposición triangular y se desplaza en las tres dimensiones del espacio merced a la articulación de los tres brazos que lo sostienen (fig. 8.2). Esta configuración triangular es precisamente a lo que alude su apelativo “delta”. Otra de las características principales es su base o cama de impresión circular, que aquí es estática. Las impresoras delta presentan una mayor velocidad de impresión y al contar con una plataforma fija, sin movimiento, pueden ofrecer mayor precisión en la impresión y permiten trabajar con materiales más fluidos, como la extrusión de arcilla. Otra ventaja de las impresoras delta es que resultan fácilmente redimensionables, sin que esto afecte a su calidad.

Dentro de los modelos de impresoras delta, el más difundido y con más alto nivel de evolución y depuración en su diseño es la impresora Kossel⁵, en sus diferentes versiones. Pese a la dificultad inicial que presentan estas impresoras para su calibración, una vez lograda se obtienen unos resultados de gran calidad. Para el presente proyecto se ha utilizado una impresora Kossel Pro debido principalmente su velocidad y su considerable volumen de impresión.



Las características más destacadas de esta impresora son:

- Velocidad de impresión: 320 mm/s en los tres ejes.
- Resolución: 100 pasos/mm en los tres ejes.
- Repetibilidad: mayor de 0,03 mm (30 micras)
- Volumen de construcción: cilíndrico, 250 mm de diámetro, 290 mm de altura.
- Estructura: triángulo, OpenBeam + esquinas impresas.
- Superficie de impresión: vidrio redondo caliente, inmóvil.
- Masa del efector final con cabezal completo: menos de 50 gramos.
- Sencillez: menos de 200 piezas.
- Calibración: completamente automática del nivel de la superficie de impresión.

Fig. 8.2. Impresora delta

⁵ Al igual que otras impresoras agrupadas en el proyecto RepRap, las impresoras Kossel llevan el nombre de un eminente científico, en este caso Albrecht Kossel, bioquímico alemán y pionero en el estudio de la genética.

Electrónica

La electrónica de código abierto que controla la gran mayoría de impresoras Kossel está basada en el conjunto RAMPS, abreviatura de “RepRap Arduino Mega Pololu Shield”. Está diseñado para integrar y manejar toda la electrónica necesaria en una impresora RepRap en un pequeño paquete y acorde con la premisa de bajo costo. RAMPS interconecta una placa controladora Arduino Mega con el conjunto de componentes electrónicos integrados en la impresora 3D, además de ofrecer mucho espacio para su expansión. Su diseño modular incluye controladores paso-a-paso enchufables y electrónica de control del extrusor en un escudo o *shield* acoplado a la placa Arduino Mega que facilita el servicio, el reemplazo de piezas, la actualización y la expansión. Esta controladora en formato shield para Arduino Mega, diseñada por la comunidad RepRap, es compartida por otros muchos modelos de impresora 3D. Puede manejar hasta 5 motores paso-a-paso mediante los controladores Pololu A4988 o DRV8825, lo que lo hace ideal para proyectos CNC en que se necesite el control de hasta 5 motores. Tiene salidas para fin de carrera, calentadores, ventiladores y termistores. Y además cuenta con conectores que permiten integrar una pantalla LCD de visualización y una tarjeta SD para lograr una máquina autónoma que no precisa estar conectada a un ordenador para su funcionamiento.

Las principales características de las controladoras RAMPS son:

- Puede controlar máquinas cartesianas, delta y extrusoras en frío (impresión de arcilla, mortero, etc).
- Tres mosfets para salidas de calefactor/ventilador y tres circuitos de termistor para el control de la temperatura.
- Funcionamiento a 5A para mayor seguridad y protección de los componentes.
- Protección de cama con calefactada con fusible adicional de 11A.
- Se adapta a cinco placas de controlador de motor paso-a-paso Pololu A4988 o DRV8825.
- Los drivers Pololu se encuentran en los enchufes del cabezal de las clavijas, por lo que pueden ser reemplazados fácilmente o extraídos para ser usados en diseños futuros.
- Todos los mosfets están conectados a pines PWM para mayor versatilidad.
- Ampliable con tarjeta SD y pantalla LCD táctil o con encoder de control.

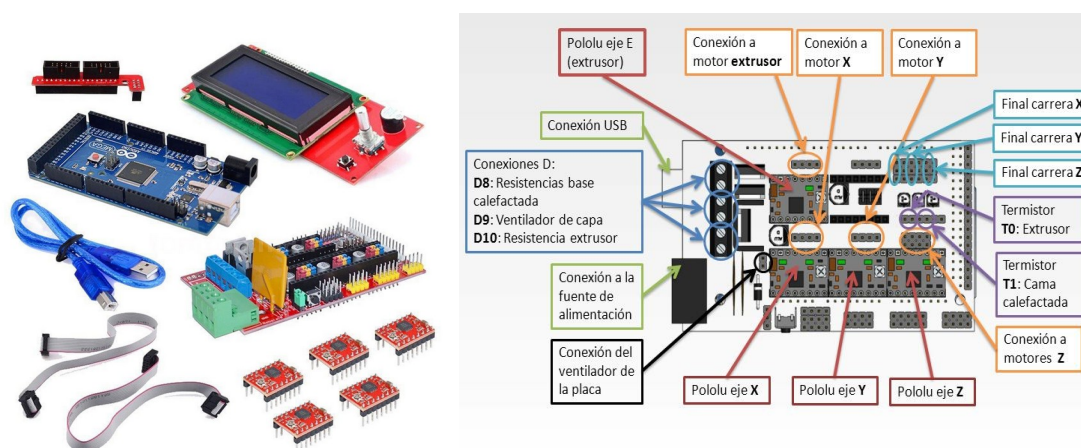


Fig. 8.3. Componentes de la electrónica RAMPS 1.4 (izqda). Conexiones de RAMPS 1.4 a los distintos componentes de la impresora (dcha)

Software de impresión

El software que permite preparar y enviar un modelo a la impresora 3D y que ésta realice los procesos necesarios para su reproducción material mediante la tecnología de deposición fundida consta de dos bloques o niveles. En primer lugar es preciso configurar y cargar en la placa controladora de la propia impresora, en el caso de la electrónica RAMPS, la Arduino Mega, un programa de control directo de la máquina o firmware⁶. Es el software que tiene una relación más directa con el hardware, siendo el encargado de controlarlo para ejecutar correctamente las instrucciones externas, y ha de estar específicamente configurado o adaptado a la electrónica y los componentes mecánicos de la impresora. De hecho, el firmware es uno de los tres principales pilares del diseño electrónico junto con el software que integra las instrucciones externas que se envían al dispositivo y el hardware.

⁶ El soporte lógico inalterable (firme, sólido) de un hardware. Se trata de un programa que establece la lógica de más bajo nivel y que controla los circuitos electrónicos del dispositivo.

re o soporte físico del propio dispositivo. En el ecosistema de código abierto podemos encontrar diversos firmwares fácilmente integrables en la electrónica RAMPS, siendo Marlin el más utilizado junto con su versión RepRapFirmware dirigida a procesadores modernos de 32 bits. Marlin fue creado para el hardware de RepRap y Ultimaker y hoy en día maneja la mayoría de las impresoras 3D del mundo, ofreciendo un control fiable, flexible y preciso. Como proyecto de código abierto Marlin se encuentra alojado en el repositorio Github, una plataforma especialmente concebida para el desarrollo colaborativo de los proyectos que alberga.

Al margen del firmware de la controladora, el otro software implicado en el proceso de impresión 3D es el específicamente concebido para editar y preparar los modelos 3D que la máquina ha de procesar. Se trata de un programa que traduce los datos contenidos en el modelo digital a lo que se denomina lenguaje máquina, un listado de instrucciones que la electrónica de la impresora puede interpretar en base al firmware que se ha cargado en ella. Los programas de impresión 3D permiten definir los parámetros de impresión, es decir la configuración relativa al posicionamiento del modelo, sus dimensiones, la resolución, material, velocidad, relleno, etc. y realizan el “laminado” o descomposición del modelo en capas según los parámetros definidos por el usuario. Como resultado se obtiene un conjunto de órdenes en lenguaje de programación de control numérico (CN) o GCode, que se pueden enviar directamente a la impresora.

Asimismo podemos encontrar programas que además de preparar y procesar el modelo nos permiten interactuar con la máquina enviando comandos específicos para la configuración del firmware, el ajuste de parámetros de impresión, la calibración, etc. mediante una conexión USB, Wifi, Bluetooth o en red. Son los programas denominados *hosts* o anfitriones, ya que integran o alojan una serie de servicios que van más allá de la configuración de los parámetros de impresión y el laminado, ofreciendo la posibilidad de un control manual de la impresora, así como la visualización del G-Code durante el proceso de impresión. Repetier-Host es uno de los programas de impresoras 3D para laminar y controlar el proceso de fabricación más populares de la comunidad RepRap. Es de código abierto y gratuito. Repetier-Host es una solución “todo en uno” que puede manejar simultáneamente diferentes filamentos y colores para permitir visualizar el resultado antes de imprimir. Repetier-Host integra cuatro laminadores diferentes -Slice3r, Slic3r Prusa Edition, CuraEngine y Skeinforge- con la posibilidad de agregar otra solución de libre elección, lo que significa que este software es compatible con prácticamente todas las impresoras 3D de deposición fundida. De este modo Repetier-Host permite controlar y ajustar la impresora 3D y envía los datos del archivo generado por el laminador alojado, o de un archivo GCode externo, para ser fabricados por la impresora 3D, ya sea mediante una conexión USB, en red o almacenando los datos en la tarjeta SD o microSD.

En presente proyecto, la preparación de las piezas a imprimir se ha llevado a cabo mediante Ultimaker Cura, un software de impresión desarrollado por uno de los principales fabricantes de impresoras 3D de deposición fundida. Cura es posiblemente el laminador 3D más utilizado en el mercado. Es de código abierto y software libre, compatible con la mayoría de las impresoras 3D de escritorio. Es capaz de leer la mayoría de los formatos de geometría 3D, como STL, OBJ, X3D, 3MF. Entre sus características más destacadas están la visualización en tiempo real del proceso de fabricación, el cálculo de la cantidad de material y el tiempo que se empleará en impresión e incluso el coste estimado. Presenta una interfaz muy intuitiva, con una amplia gama de opciones de configuración, desde el manejo básico hasta funciones avanzadas que permiten al usuario encontrar en el programa la solución que mejor se ajusta a la problemática específica de cada pieza (fig. 8.4).

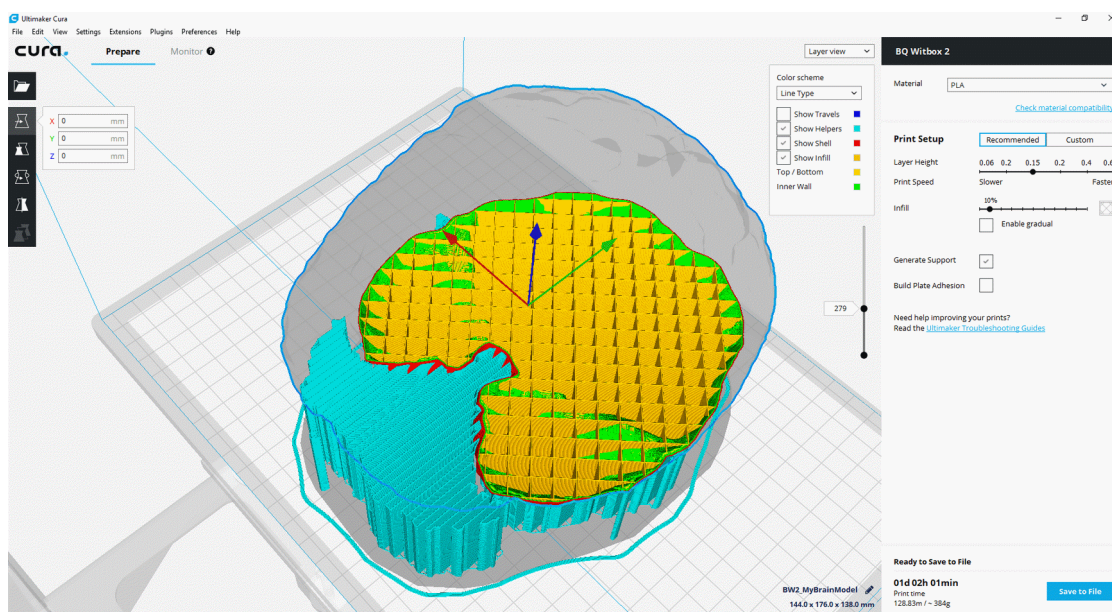


Fig. 8.4. Interfaz de Ultimaker Cura. Se puede apreciar una previsualización del resultado, el diseño y densidad del relleno, la disposición de los soportes, la estimación de tiempo y la cantidad de material que se empleará en la impresión, entre otros parámetros de impresión

Preparación del modelo para impresión

Viendo las dimensiones de la cabeza y los brazos que vamos a imprimir podemos constatar que cada uno de nuestros fragmentos es mayor que el volumen que podemos realizar en nuestra impresora 3D, lo que nos obliga a dividir cada objeto en piezas más pequeñas. Los cortes se ubicaron buscando que las uniones quedaran lo más ocultas posible, que las superficies de corte fueran planas para poder emplearlas como base de apoyo en la cama de impresión, que las áreas horizontales correspondieran a elementos menos visibles, pues el proceso de impresión las resuelve con peor calidad y finalmente, por supuesto, que proporcionen piezas adaptadas a nuestro volumen máximo de impresión. Realizamos dichos cortes en Blender mediante herramientas booleanas (fig. 8.5).

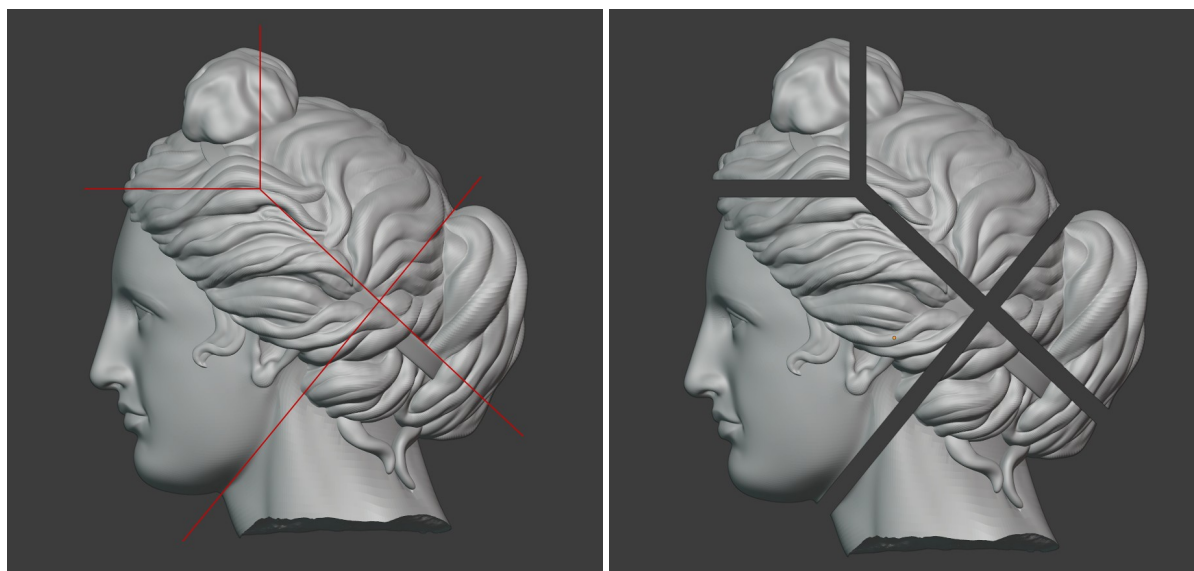


Fig. 8.5. Fragmentación de la cabeza en piezas para adaptarla al volumen máximo de impresión disponible



Fig. 8.6. Fragmentos preparados para impresión

A continuación, cada fragmento es importado en Cura, donde se fijan los parámetros que definirán la impresión, como orientación, espesor de capa, cantidad de relleno o la presencia de soportes que sujeten la estructura durante la impresión (fig. 8.7 y 8.8).

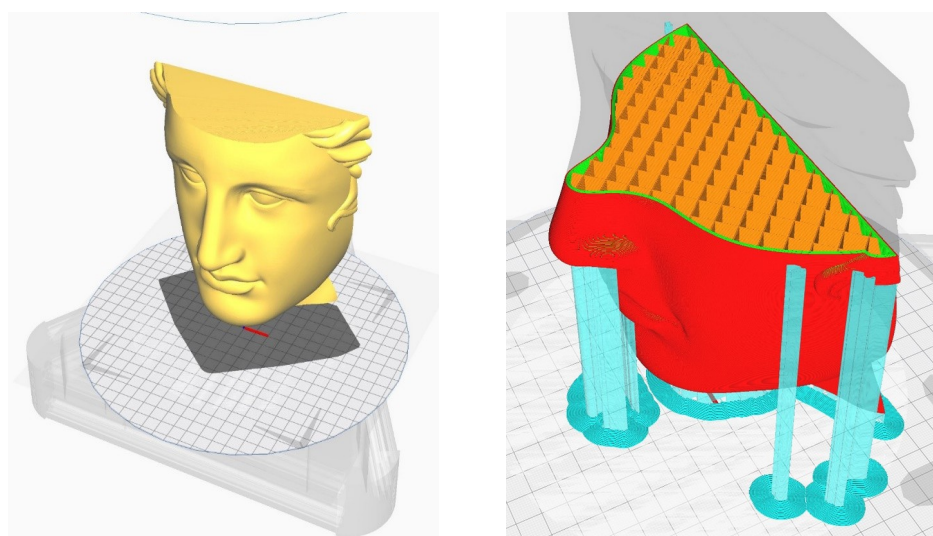


Fig. 8.7. Preparación del modelo en el entorno de Cura, posicionamiento, redimensionado y previsualización del resultado, donde se muestra en color rojo la superficie exterior de la pieza con la altura de capa definida por el usuario, en verde los perímetros interiores, en naranja el relleno y en azul los soportes y las superficies de adhesión a la cama.

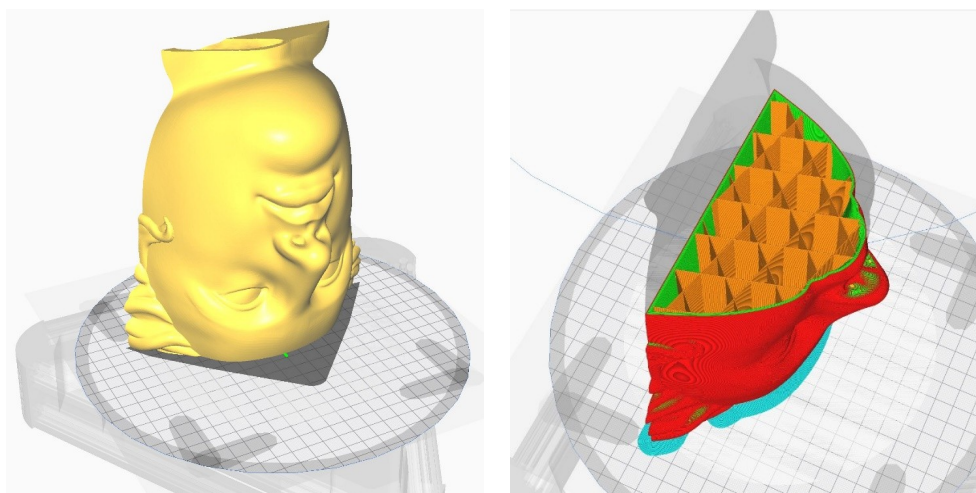


Fig. 8.8. Modificando la orientación de la pieza es posible reducir o incluso evitar la necesidad de colocar soportes



Fig. 8.9. Los cuatro fragmentos de la cabeza una vez impresos

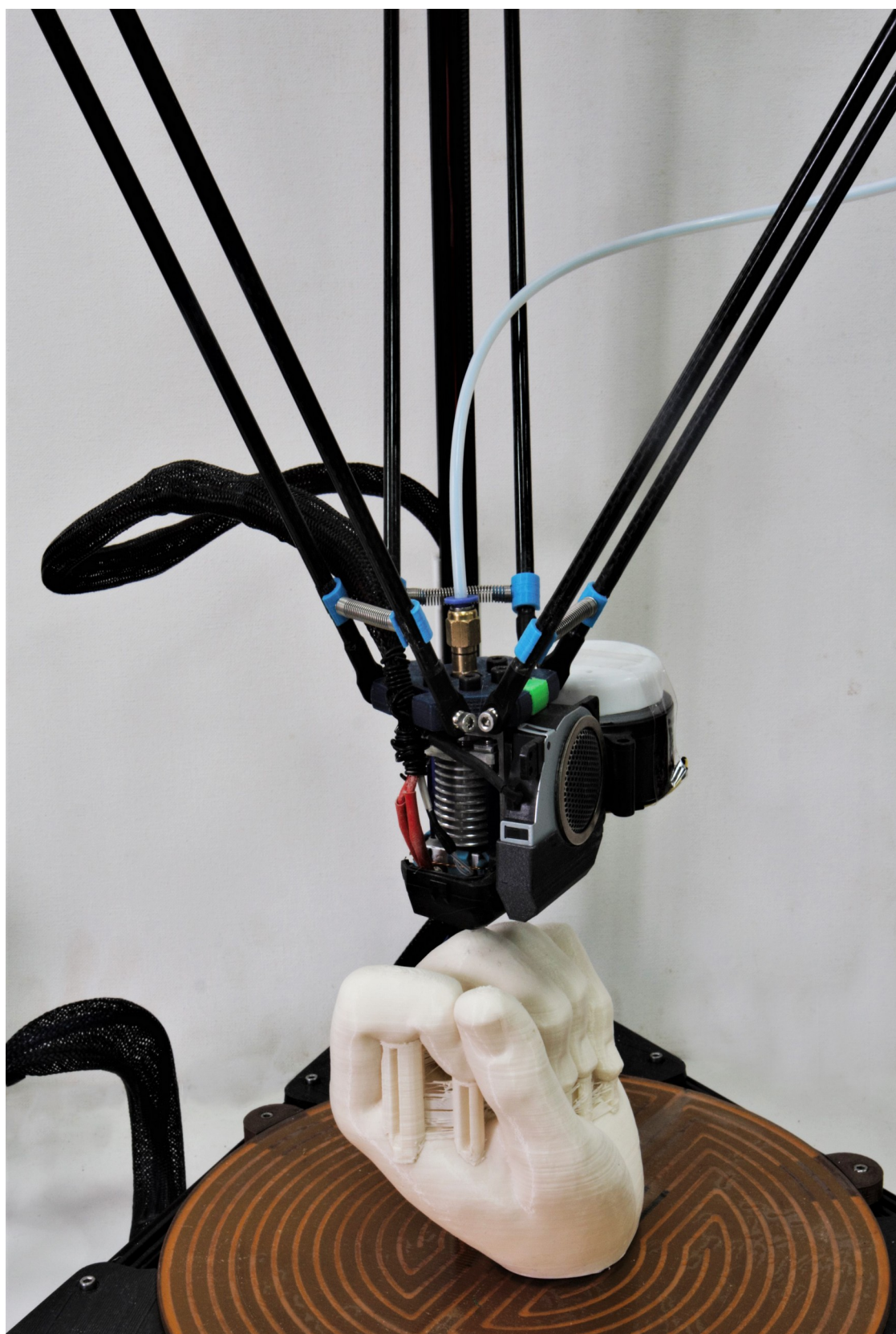


Fig. 8.10. Impresión en proceso

9. CONCLUSIÓN DEL PROYECTO

Una vez impresos todos los fragmentos y adheridos a la escultura original mediante métodos no intrusivos y fácilmente eliminables, obtenemos el resultado final de nuestra reconstrucción, planteada bajo la premisa de ser realizada íntegramente mediante software y hardware libre o gratuito. Las siguientes imágenes muestran el resultado de nuestro proyecto, corroboran nuestra hipótesis y certifican la solvencia de las actuales soluciones libres en los ámbitos de la conservación y restauración escultóricas. Nuestra experiencia, en la que hemos construido una mesa rotatoria automática sincronizada con la cámara fotográfica empleando hardware libre, hemos digitalizado una pieza escultórica de tamaño natural con Meshroom, hemos procesado los diferentes fragmentos escaneados y creado los elementos faltantes con Blender y hemos impreso dichas piezas utilizando software y hardware libre, puede fácilmente replicarse por parte de cualquier persona interesada en la digitalización de bienes culturales y en su restauración, conservación y difusión, todo ello sin apenas desembolso económico, lo que hace este tipo de intervención accesible a una innumerable cantidad de perfiles, desde profesionales, académicos, conservadores o artistas hasta estudiantes o aficionados.







Una versión online de la escultura restaurada puede verse en la siguiente web: <https://skfb.ly/ooqwZ>

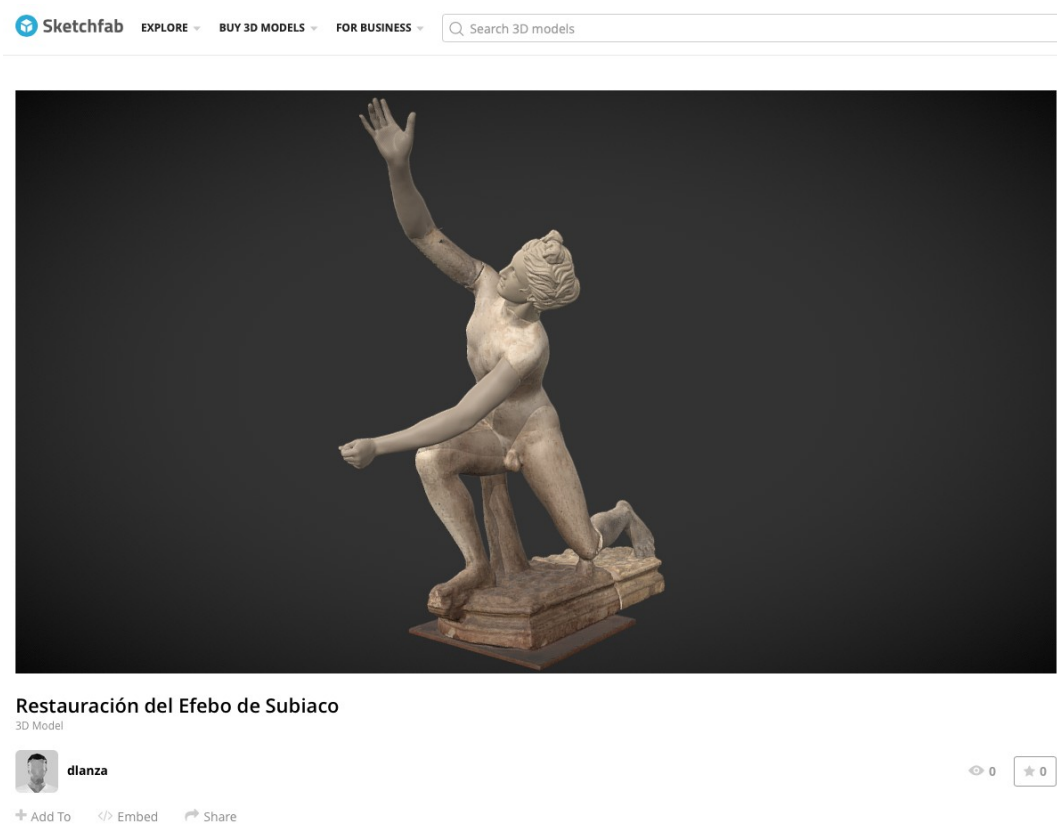


Fig. 9.1. Visualización del resultado de la restauración desde un navegador web en Sketchfab: <https://skfb.ly/ooqwZ>

10. TABLA RESUMEN DE HERRAMIENTAS LIBRES Y RECURSOS ONLINE

Para concluir el presente estudio incluimos una tabla resumen de las herramientas y recursos utilizados a lo largo de este proyecto y que pueden ser de utilidad en proyectos de captura fotogramétrica, modelado digital e impresión 3D, indicando como su estatuto como software libre o gratuito.

Recurso	Descripción	Software libre / Gratuito
Arduino Uno	Placa lógica programable	Hardware libre
RAMPS	Electrónica para impresión 3D	Hardware libre
AliceVision Meshroom	Programa restitución fotogramétrica	Software libre
Blender	Diseño 3D	Software libre
Ultimaker Cura	Impresión 3D	Software libre
Sketchfab	Visualizador 3D online	Dispone de versión gratuita

