

Data Science: Estimación del peso de un cerdo

Informe De Seguimiento 2

Marc Maldonado Lorca



1 CONTEXTO Y OBJETIVOS

Desde hace unos años el sector agrario evoluciona junto a las nuevas tecnologías emergentes, lo que se conoce como Smart Farming. Este nuevo concepto aprovecha el potencial de las TIC para abaratar tanto los costes de producción como la eficiencia de las granjas. A partir de esta idea se han ideado nuevos modelos de gestión que van desde los invernaderos inteligentes y la gestión de plagas, hasta la implementación de drones agrícolas que utilizando imágenes aéreas permiten un ahorro significativo de tiempo y mano de obra a la hora de la verificación visual de un cultivo. Además no solo es aplicable al cultivo sino que existe una interesante aplicación en el ganado.

Es por eso que el CVC trabajó en un proyecto de Smart Farming relacionado con la automatización de una granja de cerdos, donde uno de los retos pendientes era estimar el peso de los cerdos con la motivación de conocer su evolución y prepararlos para dar el peso óptimo en el matadero y así obtener la máxima rentabilidad posible. A la hora de entregar un cerdo al matadero su valor depende de lo que se aproxime su peso a los 120kg, este valor del animal disminuye tanto si su peso es inferior o superior al estándar establecido, es por eso que el control de la evolución de estos animales es tan importante para obtener beneficios.

El concepto consiste en utilizar una cámara 3D para hacer fotos cenitales de los animales y así eliminar el costoso proceso de utilizar una bascula y agilizar el pesaje aumentando de esta manera el número de cerdos pesados por unidad de tiempo [1].

Este problema ya ha sido planteado por muchas personas, instituciones y empresas, además de haber sido abordado de distintas maneras y con distintos animales. Se han utilizado imágenes 2D donde se segmenta al cerdo utilizando técnicas de morfología, se construye su esqueleto y con las medidas de este utilizando una regresión se obtiene el peso [3]. En otro estudio utilizando vacas y una reconstrucción 3D similar a nuestros datos y extrayendo diferentes atributos como la altura, la anchura, el área, etc, consiguen estimar mediante regresión el peso de la vaca entre otros factores[4]. También existe otra solución que utiliza imágenes de profundidad con las cuales utilizando redes neuronales convolucionales consigue estimar tanto el peso como otras variables biométricas del animal [5]. Finalmente otro estudio mediante nubes de puntos como las que disponemos y analizando la postura de los cerdos, analizando la relación entre volumen y postura del animal consiguen mediante regresión estimar el peso.

Además de todos estos estudios ya existen productos en el mercado variados con diferentes enfoques. La empresa PLF AGRITECH [6] ha puesto en práctica diferentes tecnologías para el desarrollo



Fig. 1: Gafas de realidad virtual para la estimación de peso

de los animales en las granjas inteligentes, aportan una solución al problema de la detección de peso además de un alimentador automático inteligente y un control sobre el impacto medioambiental de estas granjas en términos de aire. FANCOM [7] a parte de tener productos enfocados al crecimiento de otros animales como los pollos o productos destinados al cultivo de hongos también aportan una solución al control del peso de los cerdos con su producto EYEGROW [8]. Con una cámara de profundidad en el techo ellos son capaces de segmentar los cerdos individualmente y estimar el peso de ellos simultáneamente para después con la ayuda de un software poder analizar la evolución. Para esta tarea es interesante leer un artículo de SENSORS [9] que bajo el pretexto de la existencia de pocos datasets de cerdos ofrecen un modelo capaz de identificar a estos en fotos cenitales además de identificar la posición de sus distintas partes del cuerpo. Una aplicación similar a la anterior es la de GROSTAT [10].

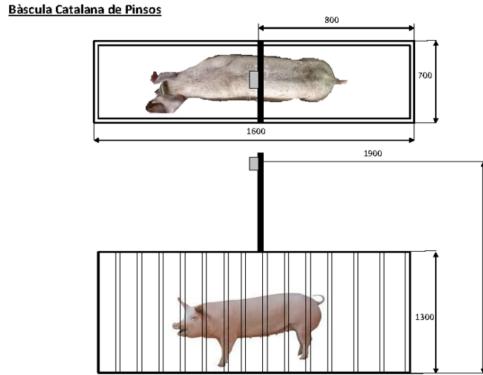


Fig. 2: Báscula con montaje de la cámara

Existe un enfoque diferente en cuanto a la manera de utilizar la cámara, mientras empresas como la anterior deciden colocar la cámara en un lugar fijo, otras empresas optan por la manipulación de la cámara de manera manual, de esta manera se facilitaría la selección del cerdo a pesar. Es el caso de H+L [11] o PIGGY CHECK [12] que mediante cámaras 3D incorporadas en dispositivos móviles consiguen estimar los pesos de los cerdos de manera individual. Existe una solución similar desarrollada por Southwest Japan Univ. [13] pero esta vez la cámara está acoplada a unas GOOGLE

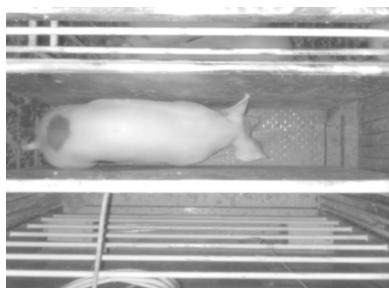


Fig. 3: Ejemplo de imagen cenital

glasses [14] como se muestra en la **Fig 1** de manera que tan solo mirando al cerdo tendríamos la información, aunque todavía no se ha lanzado al mercado. El peso es un buen indicador de la salud del cerdo pero no es suficiente, DEGREE2ACT [16] ha desarrollado un gadget para smartphone que mediante una cámara de temperatura nos informa de posibles enfermedades del cerdo, lo cual junto a una estimación del peso podría ser una potente herramienta.

Analizando el contexto y la distribución de nuestra granja y teniendo en cuenta el recorrido que realizaban los cerdos, se instaló en la báscula tradicional la cámara 3D BASTER BLAZE [2], la cual se adapta bien por su resistencia, encargada de recoger el modelo 3D de los cerdos, con imágenes como la que se muestra en la **Fig 3**, y su peso real, todo esto con la ayuda de células fotosensibles que controlarían la entrada individual de cada cerdo a la báscula, además de la instalación de un lector RFID y los respectivos PLC y PC, el montaje se muestra en las **Fig 2** para procesar y almacenar los datos obtenidos en forma de nube de puntos.

El objetivo real del proyecto extraer el potencial del dataset de las nubes de puntos utilizando modelos de machine learning y encontrar la solución más óptima posible al problema de la estimación del peso de un cerdo utilizando una cámara 3D. En un futuro esta solución combinada a distintas ideas podría ser un producto con bastante potencial.

2 METODOLOGÍA

Durante la realización de esta segunda parte del proyecto se ha proseguido con el planteamiento del informe inicial siguiendo la planificación de las fechas del TFG además de cada semana hacer reuniones individuales con el tutor para así comentar los avances y poder corregir errores. La metodología decidida en el kick-off del proyecto fue la ágil, concretamente Kanban utilizando el software de JIRA [15] que además proporciona herramientas como el diagrama de Gantt. Por el momento esta decisión parece ser adecuada y se amolda perfectamente a la hora de desarrollar el proyecto. Como se explicó en el informe inicial, siguiendo esta hoja de ruta y el método Kanban organizaremos las tareas dentro de cuatro apartados en un tablero:

- **POR HACER:** Aquí se colocarán todas las tareas que todavía no se han empezado a hacer, de aquí solo se podrán mover a la columna EN PROGRESO.
- **EN PROGRESO:** Aquí se colocarán las tareas que se encuentran en desarrollo, solo se podrán mover hacia el estado EN REVISIÓN.
- **EN REVISIÓN:** Esta columna contiene las tareas acabadas pendientes de revisión por el tutor, en el caso de completarse satisfactoriamente se moverán a HECHO, en el caso contrario volverán a EN PROGRESO.
- **HECHO:** Este apartado contendrá las tareas finalizadas.

3 PLANIFICACIÓN

Se planificaron diversas tareas en el primer informe de seguimiento que durante su ejecución han ido cambiando, eliminándose o añadiendo nuevas.

En la **Fig 4** y en la **TAULA 1** se muestran la nueva planificación de tareas que conforman el proyecto repartido por semanas. Algunas de estas tareas se han pospuesto para la última fase del proyecto, además de descartarse el uso de GNN.

La parte de la segmentación de los cerdos se podría dar por finalizada.

En cuanto a la estimación del peso se han explorado diversas soluciones. Se han descartado diversas ideas como las GNN y nos centraremos en el empleo de las CNN, transformando los datos para tratar de exprimirlos al máximo.

Es posible que durante este último tramo aparezcan nuevas ideas o se descarten otras, así que puede cambiar la planificación del proyecto.

Fase 1	Fase 2	Fase 3
Estudio Dataset Visualización datos Segmentación clásica Conversión a nube de puntos Detección de objetos Segmentación semántica Limpieza de máscaras Informe de progreso 1	Tratamiento datos peso Estudio métodos de regresión Entrenar CNN peso Informe de progreso 2 Preparación Datos Imagen	Comparativa con otras soluciones Analisis resultados Entrenar CNN peso Preparación datos images Presentación final

TAULA 1: Backlog

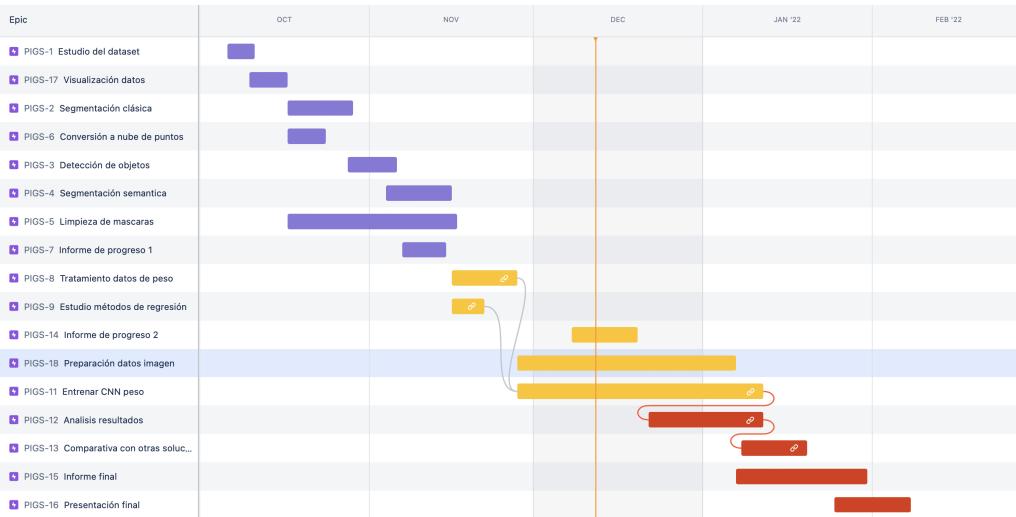


Fig. 4: Diagrama de Gantt

4 DESARROLLO

4.1 Primera etapa

Están parte del desarrollo se ha focalizado en la segmentación de los cerdos experimentando con diferentes técnicas.

Primero, al no disponer de suficiente cantidad de imágenes ni de una imagen de la báscula vacía se intentó mediante la media de todas las imágenes generar una nueva la cual se usaría como fondo, esto se hizo tanto con las imágenes de la cámara de infrarrojos como con las imágenes de profundidad, estos métodos han sido bautizados como **Mean** y **Mean 3D** respectivamente. Una vez con esta nueva imagen generada se realiza una subtracción a cada imagen de cerdos para tratar de eliminar el fondo, antes habiéndole aplicado un suavizado gausiano para tratar de conseguir un resultado más limpio. Seguidamente probando thresholds manuales o utilizando Otsu[27]. Generamos la máscara final. Haciéndolo con las imágenes de infrarrojos genera resultados notablemente mejores que con las imágenes de profundidad.

Seguidamente, se proporcionó una imagen de fondo para hacerle la subtracción, de la misma manera con las dos imágenes, estos métodos son los de **Difference** y **Difference3D**. Esta vez en vez de hacer la media de todas las imágenes realizamos la resta directamente con el fondo. De la misma manera se aplica un filtro de suavizado y con thresholds manuales y Otsu se generan las nuevas máscaras. Estas técnicas obtienen un rendimiento similar a las dos técnicas anteriores solo que a la inversa. En cuanto al uso de imágenes infrarrojas, este resultado se debe a que ahora a diferencia del anterior, utilizamos una imagen del fondo, debido a la superficie de la báscula se genera un reflejo de luz de un color similar al lomo del cerdo, así que la mayoría de los cerdos están segmentados por todo su cuerpo menos por la parte que coincide con el reflejo, esto explica la perdida de precisión.

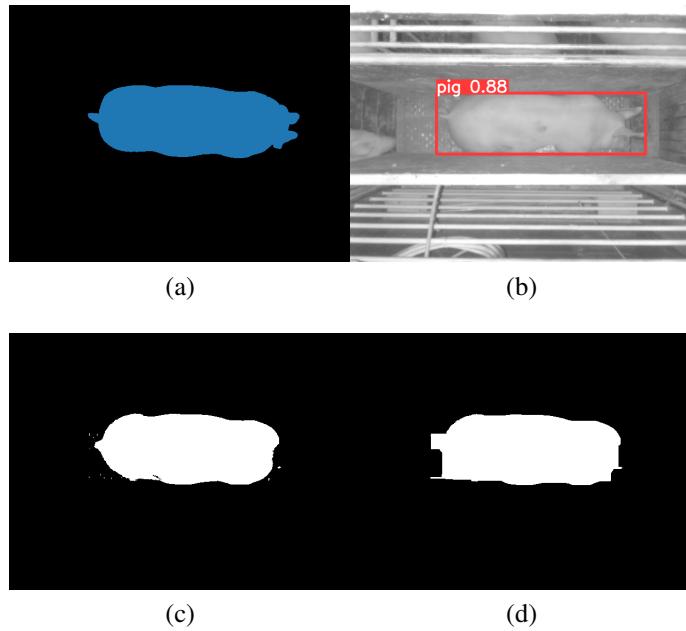


Fig. 5: Proceso de segmentación YOLO

La mejora empleando imágenes de profundidad respecto al anterior método usando las mismas se debe a que existe una diferencia en el fondo de la imagen. Aquí es donde nos dimos cuenta de que una de las paredes de la báscula cambiada de posición dependiendo de la fotografía. Esto se debe a que las paredes se ajustan dependiendo del tamaño de los animales para no permitir que entren dos simultáneamente. Aun de esta manera de todas las imágenes disponibles existen un gran número con la pared en la misma posición respecto a demás imágenes con la pared en distinta posición. Al ser la imagen del fondo una representante las imágenes que más se repiten en cuanto a posición de la pared, por eso genera tan buenos resultados.

Aquí es cuando probamos **Otsu** directamente con la imagen infrarroja, con las imágenes de profundidad queda directamente descartado, ya que segmentaría muchas partes de la báscula que para eliminarlas después utilizando morfología sería una tarea muy difícil. Hacerlo directamente con Otsu funciona bien pero no mejor que **Difference3D**.

En este punto se puede ver que el verdadero problema se encuentra con el fondo y no con los cerdos así que se busca una solución, la primera es utilizar Canny y Hough[28] para detectar las líneas de la báscula y así eliminar el problema de las paredes móviles y el fondo. Esta técnica funciona bien con el primer conjunto de imágenes, al tener nuevas se vuelve un reto imposible debido a la variedad de fotos, de este modo, se descarta y se opta por un recorte horizontal de la imagen que elimine fondo, pero no interfiera ni recorte en ningún caso alguna parte del cerdo, seguidamente aplica Otsu sobre el recorte. Hasta el momento esta técnica bautizada como **Otsu_crop** se convierte en la mejor técnica para segmentar.

También se experimenta usando Niblack y Sauvola [31], pero no se obtienen ningún resultado a apreciar.

Viendo el potencial de recortar la imagen se decide generar groundtruths para entrenar un modelo de segmentación semántica, para esto se utiliza la herramienta Hasty.ai [29] la cual proporciona diversas herramientas para el etiquetado de imágenes, a nivel semántico, de objeto y de imagen. En este caso se hizo semanticamente, ya que a partir de aquí también podemos generar bounding boxes para la detección de objetos. Esta herramienta permite etiquetar de manera simultánea con demás personas y durante el etiquetado entrena modelos que te ofrecen una posible máscara la cual tú puedes editar. Con este groundtruth se extrajeron las bounding boxes y usando Google Colab se entrenó un modelo **YOLO**[30], concretamente V5, con el cual se recorta al cerdo y se aplica Otsu sobre la imagen recortada. Genera un resultado similar a Otsu_crop, pero ligeramente peor en cuanto accuracy, de

Difference	0.47
Difference3D	0.87
Mean	0.81
Mean3D	0.36
Otsu	0.69
Otsu_crop	0.97
Yolo	0.97

TAULA 2: Accuracy métodos segmentación



Fig. 6: Ejemplo de point cloud con pared



Fig. 7: Ejemplo de point cloud correcto

todas maneras elegimos Yolo porque la diferencia es ínfima y nos asegura que siempre cogemos al cerdo y no otros cerdos que asoman la cabeza por la puerta como se puede ver en alguna imagen. Además, sobre estas máscaras generadas se aplica técnicas de morfología para llenar huecos dentro de los cerdos generados por las manchas de estos y kernels horizontales para tratar de dilatar horizontalmente la máscara y recuperar puntos pertenecientes al cerdo no segmentados. [30] para con su detección tener una imagen recortada del cerdo y aplicar Otsu sobre él. El proceso de segmentación se puede ver en la **Fig 5** siendo **(a)** el groundtruth, **(b)** la detección de Yolo, **(c)** Otsu aplicado sobre la detección de Yolo y **(d)** aplicando la morfología.

También se utilizó el groundtruth generado y un script de data augmentation para entrenar un modelo de segmentación semántica con ResNet50 [19], que debido a falta de potencia de cómputo no se ha podido realizar.

Desde el CVC usando las máscaras se han creado modelos de segmentación bastante eficaces como para substituir el método con Yolo, y como ya se ha comentado se está trabajando actualmente en una MobileNetV2 para intentar crear un nuevo método de segmentación propio, todo esto utilizando GoogleColab que presta potencia de cómputo de la que no disponemos.

Los accuracy de todos los métodos se pueden consultar en la **TABLA 2**.

Finalmente, se construye el modelo en 3D con open3D [18] y se eliminan todos los puntos que se encuentran más abajo de un determinado lindar para así eliminar suelo, seguidamente se eliminan



Fig. 8: Máscaras generadas por Segmentación semántica

Método	Acc
YOLO	0.97
Segmentación con cabeza	0.99
Segmentación sin cabeza	0.99

TAULA 3: Resultados distintos métodos de regresión

outlayers de manera estadística. Sin embargo, esta técnica debe mejorarse, en algunos casos no se detecta correctamente toda la superficie del cerdo, en otros casos se seleccionan puntos pertenecientes a la pared de la báscula como en la **Fig 6**. Otro aspecto a destacar serían las cabezas de los cerdos, algunos de ellos tienen la cabeza levantada y se segmenta su cabeza, otros la tienen agachada y no se segmenta correctamente. Un ejemplo nube de puntos sin errores sería la que se muestra en la **Fig 7**.

4.2 Segunda etapa

Esta parte del desarrollo se ha focalizado en la regresión del peso de los cerdos y la finalización del proceso de segmentación de los cerdos.

4.2.1 Segmentación

Este segundo tramo del proyecto comenzó con la finalización de la segmentación semántica de los cerdos. En el anterior informe de progreso se comenta que se preparó para entrenar una U-Net[21] con el *backbone* de ResNet-50[19], además de contemplar la opción de utilizar un modelo más ligero como MobileNetV2[20]. Finalmente, tras distintas pruebas con diferentes *backbones* y distintas configuraciones de los hiperparámetros, se consiguió un modelo que entrenado con las imágenes captadas por la cámara infrarroja y las máscaras generadas con HastyAI[29] obtenía unas métricas del 0.98 IOUScore[22] y 0.99 de accuracy usando como backbone InceptionV3[23]. Todo esto utilizando las imágenes recortadas manualmente para eliminar información que nunca será importante para la regresión y agilizar el entrenamiento del modelo. Más adelante en el desarrollo se decidió utilizar únicamente las imágenes de profundidad para entrenar el nuevo modelo de segmentación. No solo eso, sino que se generaron las mismas máscaras pero esta vez sin la cabeza de los cerdos para comprobar que la segmentación de esta parte del cuerpo del animal no sea una dificultad a la hora de estimar el peso. Se pueden consultar las métricas de la segmentación en la **TAULA 3**. Con estas métricas descartamos los métodos tradicionales planteados en el apartado utilizar este nuevo planteamiento

4.2.2 Regresión

Con esta capacidad de recortar las imágenes el siguiente paso fue utilizarlo para extraer la nube de puntos y construir una malla triangular para su posterior uso en las GNN. Para la tarea de construir las mallas se usó la librería Open3D[18] que proporciona distintas funciones para la reconstrucción de superficies: *Alpha shapes*, *ball pivoting* y *Poisson surface reconstruction*. Ajustando los parámetros de la función Poisson finalmente se consiguió la reconstrucción que se muestra en la **Fig 9**.

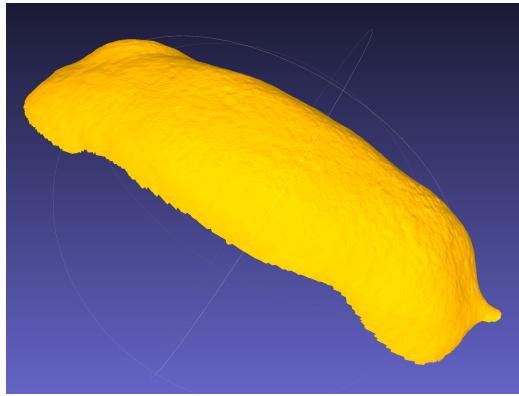


Fig. 9: Malla generada con Poisson surface reconstruction

Método	MAE
Imágenes raw	7.6
Segmentación con cabeza	4.5
Segmentación sin cabeza	4.6
Centro de masas centrado	???

TAULA 4: Resultados distintos métodos de regresión

Con estas nubes de puntos y estas mallas generadas se intentó buscar una solución al problema de regresión, pero se descartó debido a su complejidad y al escaso desarrollo existente de estas tecnologías en estos ámbitos. Se intentó adaptar el modelo de clasificación de PointNet[24] para regresión, no obstante no fue posible.

Seguidamente se procedió a mapear las imágenes con los pesos correspondientes en el archivo *xlsx*. Cada imagen consta con una serie de dígitos en el nombre que indican un *timestamp* de cuando se tomó la fotografía. Estos instantes de tiempo no encajan con el tiempo que marca la entrada de los pesos en el archivo Excel, existe una diferencia de 4 min de delay. Con esta información se genera un archivo *cvs* con el mapeo.

Se hicieron las primeras regresiones utilizando CNN con la ayuda de la librería *ktrain*[25]. Esta dispone de un *early-stop* y ajustes automáticos del *learning rate*, muy útiles para este proyecto, ya que al usar *Google Colab* tenemos ejecuciones limitadas y de esta manera las conseguimos exprimir cada ejecución al máximo para sacar partido a la ventana de computación que nos presta este servicio. Primeo se experimentó con las imágenes de profundidad en crudo y buscando mejorar estos resultados, surgió la idea de utilizar las máscaras de la segmentación sobre estas imágenes para eliminar la información irrelevante relacionada con el fondo. Con los modelos entrenados, se revisaron las imágenes con un MAE[26] más elevado y se llegó a la conclusión de que posiblemente fueran las cabezas las que aportaban una gran varianza a estos resultados, de forma que los cerdos con la cabeza más escondida tendían a pesar menos y de manera contraria los cerdos con la cabeza más expuesta pesaban más. Para aplicar esta corrección se utilizan las máscaras de los cerdos sin cabeza comentados en el apartado anterior. Las máscaras de los cerdos se pueden observar en la **Fig 8** donde (a) es la máscara teniendo en cuenta la cabeza y (b) sin esta. Durante el desarrollo también surgió la idea de entrenar un modelo de clasificación capaz de clasificar cerdos según su tamaño, pequeños medianos y grandes, para así después crear modelos regresivos individuales para cada conjunto. Este modelo de clasificación obtiene un 84% de accuracy, y con los modelos regresivos generados posteriormente a la clasificación podemos ver la mejora del MAE en los grupos de cerdos pequeños y grandes, en cambio, los medianos, al contener más clasificaciones erróneas, tanto de los pequeños como de los grandes, obtenemos un MAE bastante peor, que en promedio no mejora la estrategia de utilizar el conjunto de todas las imágenes para la regresión, así que por ese motivo se descartó este plan. En la **TAULA 4** se pueden consultar las métricas de los distintos métodos hasta el momento.

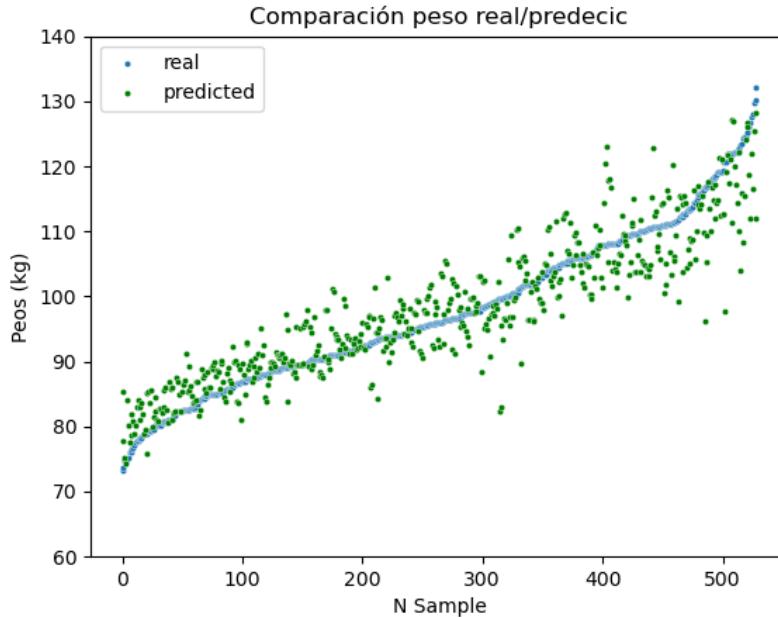


Fig. 10: Comparación predicciones y peso real utilizando máscaras completas

En la **Fig 10** correspondiente al método de segmentación con la máscara del cerdo completa podemos observar como las predicciones tienden a crear una linea horizontal en el plano, es decir, predecir un peso mayor para los cerdos pequeños y uno menor para los grandes, se seguirá trabajando con esta información.

En este punto del proyecto es todavía muy temprano para extraer conclusiones. Todavía hay puntos pendientes como entrenar el modelo aplicando centrando el centro de masas de la imagen para facilitar la tarea al regresor, entre otras ideas que puedan surgir. Además, parece haber discrepancias en la manera en como están representadas las imágenes que se tomaron antes del 6 de mayo con las posteriores, se han descartado estas para el desarrollo. Comentar también que el resultado actual de 4.5 MAE no llega a ser del todo representativo porque para ello deberíamos utilizar *cross-validation*, pero debido al problema recurrente de no disponer de tiempo ni de poder de computación para ello, nos quedamos con este resultado y confiamos que al ser tan similares los inputs, el valor del de la validación es bastante correcto. Además también se han detectado algunas anomalías en los Excels del peso de los cerdos que todavía está por resolver.

REFERENCIAS

- [1] AHDB Pork "*Pigs 2022 - Pig weighing systems*" [Online 30/9/2021] <https://www.youtube.com/watch?v=3CuLNhFOjk>
- [2] Basler "*Basler blaze*" [Online 30/9/2021] <https://www.baslerweb.com/en/products/cameras/3d-cameras/basler-blaze/>
- [3] K. Kollis, C.S. Phang, T.M. Banhazi and S.J. Searle "*Weight estimation using image analysis and statistical modelling: A preliminary study*" [Online 30/9/2021] https://www.researchgate.net/publication/236886740_Weight_Estimation_Using_Image_Analysis_and_Statistical_Modelling_A_Preliminary_Study
- [4] Dorota Anglart "*Automatic estimation of body weight and body condition score in dairy cows using 3D imaging technique*" [Online 30/9/2021] https://stud.epsilon.slu.se/6355/1/anglart_d_140114.pdf

- [5] Jianlong Zhang, Yanrong Zhuang, Hengyi Ji, and Guanghui Teng "*Pig Weight and Body Size Estimation Using a Multiple Output Convolutional Neural Network: A Fast and Fully Automatic Method*" [Online 30/9/2021] https://www.researchgate.net/publication/352102505_Pig_Weight_and_Body_Size_Estimation_Using_a_Multiple_Output_Regression_Convolutional_Neural_Network_A_Fast_and_Fully_Automatic_Method
- [6] PLF Agritech "*PLF Agritech*" [Online 30/9/2021] <http://plfag.com/technologies/>
- [7] Fancom "*Fancom*" [Online 30/9/2021] <https://www.fancom.com>
- [8] Fancom BV "*eYeGrow - Weight monitor for finishers*" [Online 30/9/2021] <https://www.youtube.com/watch?v=CSuWWgY43PA>
- [9] Eric T. Psota, Mateusz Mittek, Lance Pérez and Ty B Schmidt "*Multi-Pig Part Detection and Association with a Fully Convolutional Network*" [Online 30/9/2021] https://www.researchgate.net/publication/352102505_Pig_Weight_and_Body_Size_Estimation_Using_a_Multiple_Output_Regression_Convolutional_Neural_Network_A_Fast_and_Fully_Automatic_Method
- [10] GroStat "*GROWTH SENSOR*" [Online 30/9/2021] http://grostat.com/growth_sensor.php#prettyPhoto
- [11] H+L "*optiSCAN*" [Online 30/9/2021] https://hl-agrar.de/en_gb/optiscan/
- [12] Agro Napló "*Piggy Check – weigh pigs with your tablet PC*" [Online 30/9/2021] <https://www.agronaplo.hu/nagyvilag/piggy-check-weigh-pigs-with-your-tablet-pc>
- [13] The Mainichi "*Southwest Japan univ. develops smart glasses to visually estimate pigs' weight*" [Online 30/9/2021] <https://mainichi.jp/english/articles/20210601/p2a/00m/0na/007000c>
- [14] Glass "*Glass*" [Online 30/9/2021] <https://www.google.com/glass/start/>
- [15] Altlassian "*Jira*" [Online 30/9/2021] <https://www.atlassian.com/es/software/jira?>
- [16] degree2act "*degree2act*" [Online 30/9/2021] <https://www.degree2act.com/>
- [17] Hasty.ai "*Hasty*" [Online 10/11/2021] <https://hasty.ai>
- [18] Open3D "*Open3D*" [Online 10/11/2021] <http://www.open3d.org>
- [19] Kaiming He, Xiangyu Zhang, Shaoqing Ren and Jian Sun "*Deep Residual Learning for Image Recognition*" [Online 10/11/2021] <https://arxiv.org/pdf/1512.03385.pdf>
- [20] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov and Liang-Chieh Chen "*MobileNetV2: Inverted Residuals and Linear Bottlenecks*" [Online 10/11/2021] <https://arxiv.org/pdf/1801.04381.pdf>
- [21] Olaf Ronneberger, Philipp Fischer and Thomas Brox "*U-Net: Convolutional Networks for Biomedical Image Segmentation*" [Online 14/12/2021] <https://arxiv.org/pdf/1505.04597.pdf>

- [22] pyimagesearch "Intersection over Union (IoU) for object detection" <https://www.pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection> [Online 14/12/2021]
- [23] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens and Zbigniew Wojna "Rethinking the Inception Architecture for Computer Vision" [Online 14/12/2021] <https://arxiv.org/pdf/1512.00567v3.pdf>
- [24] Charles R. Qi, Hao Su, Kaichun Mo and Leonidas J. Guibas "PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation" [Online 14/12/2021] <https://arxiv.org/pdf/1612.00593.pdf>
- [25] amaiya "amaiya/ktrain" [Online 14/12/2021] <https://arxiv.org/pdf/1612.00593.pdf>
- [26] Statistics How To "Absolute Error Mean Absolute Error (MAE)" [Online 14/12/2021] <https://www.statisticshowto.com/absolute-error/>
- [27] LearnOpenCV "Otsu's Thresholding with OpenCV" [Online 10/11/2021] <https://learnopencv.com/Otsu-thresholding-with-opencv/>
- [28] LearnOpenCV "Hough Transform with OpenCV(C++/Python)" [Online 10/11/2021] <https://learnopencv.com/hough-transform-with-opencv-c-python/>
- [29] Hasty.ai "Hasty" [Online 10/11/2021] <https://hasty.ai>
- [30] Yolov5 "ultralytics/yolov5" [Online 10/11/2021] <https://github.com/ultralytics/yolov5>
- [31] scikit-image "Niblack and Sauvola Thresholding" [Online 10/11/2021] <http://www.open3d.org>