

Resultados obtenidos

Núm	Arquitectura	Descripción y cambios comparando con la prueba anterior	Resultados	Código Disponible	Dice entrenamiento	Dice test
1	Unet sin backbone	arquitectura básica sin backbone. Datos de entrada: 1620 imágenes 2D convertidas de nii.gz a png. Normalización al rango [0,1] y Data Augmentation aplicados..	Solo permite batch size muy bajo: 2, debido a las limitaciones de recursos computacionales	Anexo 1	0,4845	0,4754
2	Unet con ResNet	Utiliza ResNet50 pre-entrenado como base con conexiones skip. Datos de entrada preprocesados según requisitos de ResNet. Normalización al rango [0,1] y Data Augmentation aplicados.	Solo permite batch size muy bajo: 2	Anexo 1	0,483	0,4731
3	Unet con EfficientNet	Utiliza EfficientNet pre-entrenado como base con conexiones skip. Datos de entrada preprocesados según requisitos de EfficientNet. Normalización al rango [0,1] y Data Augmentation aplicados.	Solo permite batch size muy bajo: 2	Anexo 1	0,4857	0,4767
4	SALT	Modelo pre-entrenado para segmentar 145 estructuras corporales. Datos de entrada: 1 imagen nii.gz para prueba. Transformación al formato channel first aplicada.	No funciona bien para la segmentación de mandíbula y dientes, porque no tiene clases tan específicas, sino huesos, órganos y tejidos. Los resultados de test visualmente están muy lejos del nivel óptimo.	Anexo 2	NA	NA
5	nnUnet	Datos de entrada: 46 imágenes médicas 3D .mha sin transformación, 42 clases de etiquetas originales (background, cada diente, mandíbula), 1000 épocas preestablecidas por el modelo	Se interrumpe ejecución por falta de RAM	Anexo 3	NA	NA
6	nnUnet	Datos de entrada: 5 imágenes médicas 3D .mha sin transformación, número de clases de etiquetas sin cambio. El problema de 1000 épocas se resuelve introduciendo una nueva clase de entrenador que hereda propiedades de la clase original sobrescribiendo el número de épocas. Se utiliza 1 época para debugging. Nueva clase: <code>class nnUNetTrainer_1epoch(nnUNetTrainer).</code>	1 época dura más de 1 hora y para algunos folds de cross-validation no se finaliza	-	NA	NA
7	nnUnet	Datos de entrada: 2 imágenes médicas 3D .mha. Se introduce transformación de imágenes de 3D a 2D utilizando código en Python dividiendo en slices: 274 slices en formato .nii.gz para cada imagen mha. En vez de 42 clases, se utilizan solo 4 juntando clases de menor interés, se utiliza técnica de Region-based training. Entrenamiento sigue con 1 época.	Se finalizó el entrenamiento con Dice, se realizó la predicción con la imagen de test	Código de transformación Anexo 4, Código de entrenamiento Anexo 5	0,128877	0,161049
8	nnUnet	Épocas aumentadas a 20 utilizando nueva clase: <code>class nnUNetTrainer_20epoch(nnUNetTrainer)</code> , aumentado número de imágenes de entrada de 2 a 6 imágenes .mha, lo que equivale a 1620 imágenes 2D. Preprocesamiento según requisitos de nnUnet	Mejores resultados de todas las Arquitecturas	Código de transformación Anexo 6, Código de entrenamiento Anexo 7	0,659006 promedio, 0,977265 mandíbula	0,660252 promedio, 0,953292 mandíbula

Como ejemplo, en la ilustración se pueden observar la imagen original, la etiqueta original y la etiqueta predicha por la última arquitectura 8 de nnUNet:

