Preprocessing of images

DICOM (Digital Imaging and Communication in Medicine) is the most common and international standard for medical image storage, transmission, processing, and visualization. DICOM objects include, in addition to the images themselves, a large amount of metadata, from study acquisition parameters to patient demographic information, which means that the de-identification of this type of image can be complex and attention must be paid to the context in which its acquisition and subsequent processing takes place.

The software recommended for the de-identification of the DICOM files is DICOM Confidential [1], a freely available data anonymization tool for imaging. It is a Java-based de-identification toolkit that enforces confidentiality policies.

After the de-identification, images must be preprocessed to become interpretable by Deep Learning systems. However, there is still a lack of consensus on what preprocessing to do.

In the present document, DICOM files with an image size of 512 x 512 pixels and a variable number of slices were used, all of them corresponding to head-CT studies without intravenous contrast. The preprocessing method used consisted of the following steps.

Step 1 Reading the pixel values of image slices and scaling them to Hounsfield Units (HU)

First, the pixel values (PixelArray) of images corresponding to the different slices were read and these were ordered according to the DICOM ImagePositionPatient attribute, which contains the coordinates of the upper left corner (center of the first voxel) of each slice.

When visualizing the images, it was observed that in some of them the circular limit of the field of view was appreciated. Further, in the histogram of these images, a peak around the value -2000 was observed. This peak was not present in images in which the limit of the field of view was not appreciated.

The pixel values of images (PixelArray) were not expressed directly in Hounsfield Units (HU). This meant that, in order to make measurements or comparisons, it was necessary to first scale them to HU. For this, the values of the slope (Rescale.Slope) and the intercept (Rescale.Intercept) of the linear transformation between the pixel values stored in the PixelArray and the HUs were used, both contained in the DICOM header. With these two values, after extracting the PixelArray from the files, the original values of the pixels were rescaled to HU.

The value of the pixels that were outside the limit of the field of view (outside-of-scan pixels) was set by convention to -1000 UH (the value that corresponds to the air).

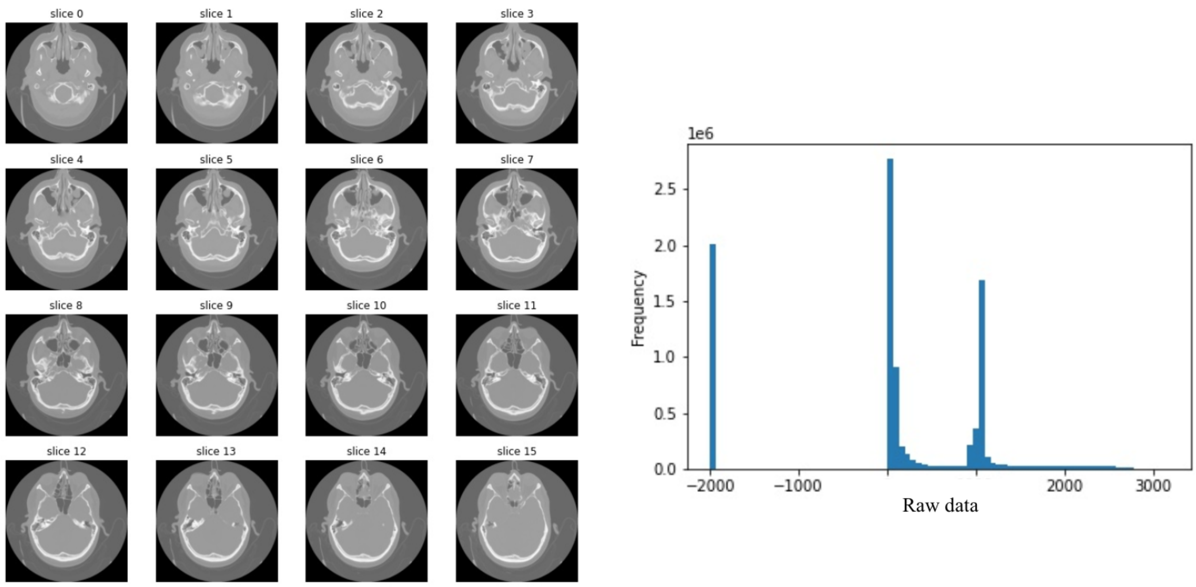

Figure 1 Example showing both the images (not all images are included) and the histogram of one of the patients before applying any preprocessing technique. A peak is observed in the histogram around the value -2000, which corresponds to the pixels outside the circular limit of the field of view.

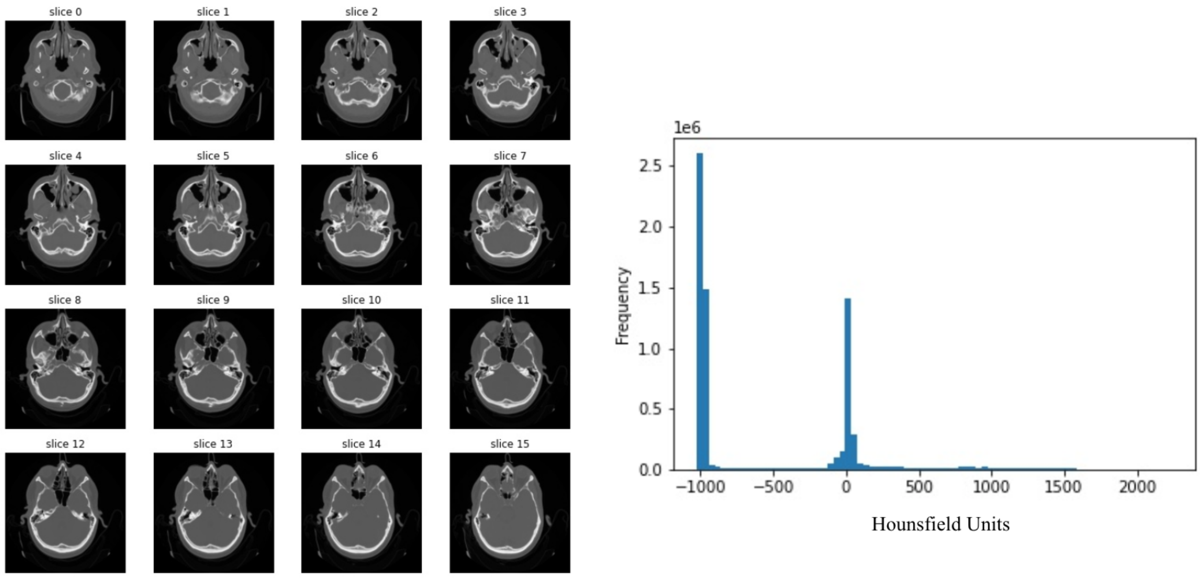

Figure 2 The same patient of Supplementary Figure 1 after performing the transformation to HU. The circular limit of the field of view is no longer visualized in the images, and in the histogram, instead of the peak at -2000, a peak at -1000 HU is observed, corresponding to the pixel value of the air surrounding the brain.

Step 2 Resampling

CT image is made up of voxels that may not have the same dimensions in the three directions of space (anisotropic) and may vary in size from one series to another. Deep learning algorithms work better with voxels that are isotropic and of the same size between the different series. Therefore, resampling is done to obtain isotropic voxels.

Pixel spacing (PixelSpacing) is the distance between the centers of two adjoining pixels. Generally, this distance is equal to the size of the pixel (to its width or height, because they are usually square), but these two concepts do not always coincide. In fact, sometimes the pixel size is smaller than the pixel spacing. Nonetheless, in our case, the pixel spacing was equal to the pixel size.

The resampling factor was obtained by adding the pixel size (0.488281 mm) to the slice thickness (2535, 4.5 or 5 mm, depending on the patient) so that, in cases were slice thickness was different from the pixel size, the resulting voxel would always be isotropic and its size would be equal to the sum of both distances. The reason the voxel needs to be larger is because it cannot be resampled to a higher resolution than the original (Nyquist-Shannon Sampling Theorem [2]). Because of this, we inevitably worsen the resolution of our image with this resampling method.

Step 3 Windowing and normalization

Given that CT has the capacity to present more levels of gray than the human eye is capable of distinguishing, we visualize images applying different windows, that is, HU ranges that are adjusted to take advantage of the entire dynamic range of the monitor improving the contrast of the regions of interest.

In this third step, a window (15-100 HU) was applied, and then the values of the selected pixels were normalized to values among 0 and 1. This window removes the bone (because it has higher HU) and "highlights" the ICH.

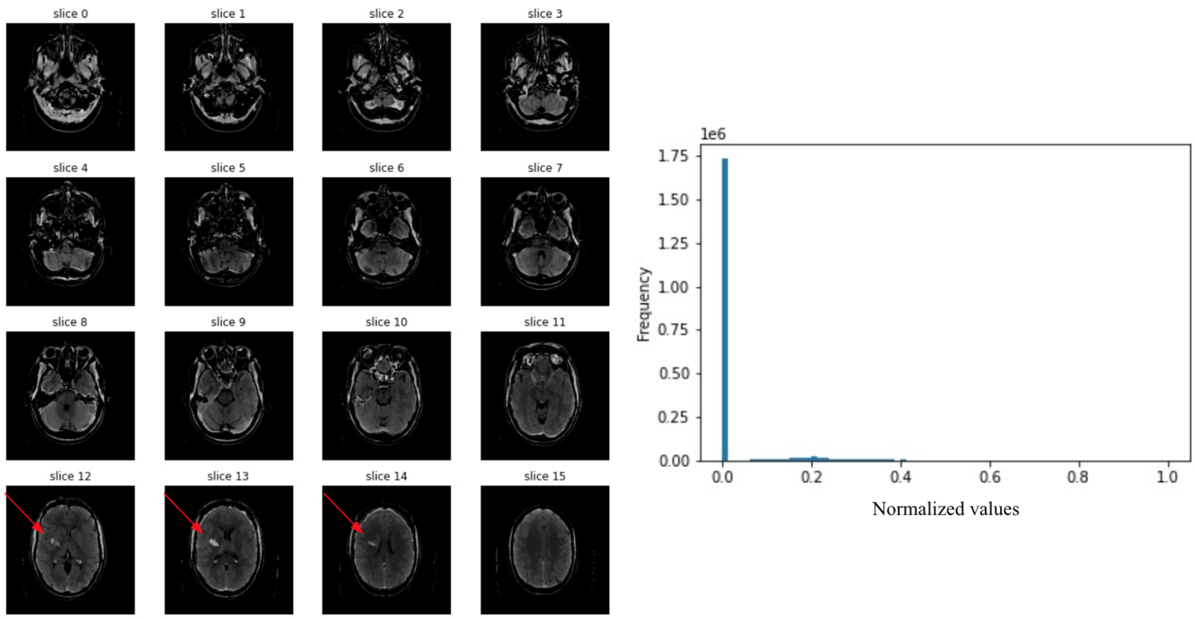

Figure 3 The same scan of Supplementary Figures 1 ans 2 previous figures after having applied the window and having normalized the values between 0 and 1. ICH is identified in the right basal ganglia (red arrow).

Step 4 Resizing images

All archives included in this study had the same slice size, 512 pixels x 512 pixels. We reduced the size of the images to 128 pixels x 128 pixels, to reduce the computational cost, using a standard spline interpolation method.

Step 5 Number of slices per patient (z-axis)

Finally, the slice number of each scan was equalized. For this, since the maximum number of slices of scans was shown to be 45, all the scans were equalized to 45 slices by adding empty slices (black slices) to the top of the scan.

References

1. Rodríguez González, D., Carpenter, T., van Hemert, J. I., & Wardlaw, J. (2010). An open source toolkit for medical imaging de-identification. European radiology, 20(8), 1896–1904. https://doi.org/10.1007/s00330-010-1745-3
2. Shannon, C.E. (1949). Communication in the presence of noise. Proceedings of the Institute of Radio Engineers. 37 (1): 10–21. https://doi.org/10.1109/jrproc.1949.232969