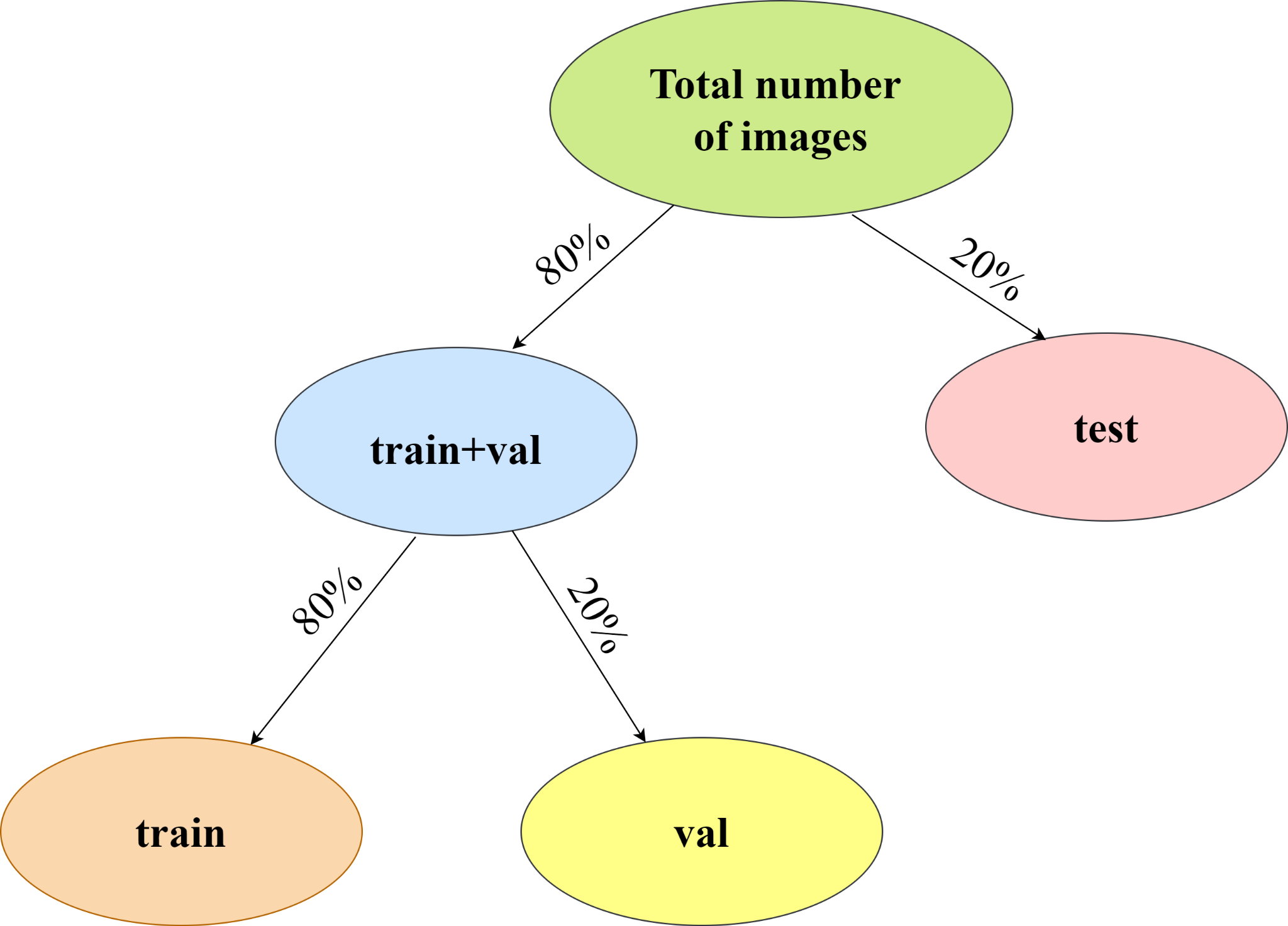
**Step-1: Collecting your dataset:-**

For now, we have a small image dataset. The dataset consists of 426 images of original and copied images. The distribution among the two categories are as follows:

| **Category** | **No. of images** |
| --- | --- |
| Original | 224 |
| Copied | 202 |
|  |  |



| **Category** | **Total number of images**  **(T)** | **No. of training images**  **(0.8\*0.8\*T)** | **No. of validation images**  **(0.2\*0.8\*T)** | **No. of testing images**  **(0.2\*T)** |
| --- | --- | --- | --- | --- |
| Original | 224 | 143 | 35 | 46 |
| Copied | 202 | 130 | 32 | 40 |
|  |  |  |  |  |

The above is the illustration of the folder structure. The training dataset folder named “train” consists of images to train the model. The validation dataset folder named “Val”(but it is shown as validation in the above diagram only for clarity. Val refers to this validation dataset). The images of the folder named “test” are completely unseen images.

The performance of our model on the testing dataset shows how accurate our model is.

**Step-2: Pre-processing of the images:-**

We will be training a VGG-19 model on our custom training dataset to classify among the two categories-original and copy. The pre-trained CNN model inputs an image of dimensions 224×224 of one of the two hand gestures. However, all the images of the dataset are of dimensions 300×300. Hence, they must all be resized to the required dimension.

**Step-3: Model training:-**

This step includes model building, model compilation, and finally fitting the model.

* To customize the model, we have to change its last layer alone according to the number of classes in our problem. As we have only two categories, we can code it this way
* We should use sparse categorical cross-entropy as our loss function. We will use the best optimizer called adam optimizer as it decides the best learning rate on its own.
* We must ensure that our model does not get overfit during the training

**Step-4: Model evaluation:-**

* To evaluate our model by testing it on the test dataset.
* To call our deep learning model good and efficient, it is not only enough to look at its accuracy but it is also equally essential to observe its classification report and confusion matrix.