# CHAPTER III
# RESEARCH METHODOLOGY

In this chapter, we present the methodology for our research project, which is divided into two main parts: image screening and DR grading. The image screening process involves the detection of ocular structures, specifically the optic disc and macula, in retinal fundus images and the screening algorithm to identify the medically suitable retinal image. The detection is achieved through the application of a correlation filtering technique. The second part focuses on DR grading, where we utilize deep learning techniques to classify the severity level of diabetic retinopathy.

## 3.1 Dataset

- **IDRiD dataset (Porwal et al., 2018):** This dataset has been curated to support various challenges related to diabetic retinopathy (DR) such as lesion segmentation, DR and diabetic macula edema (DME) severity grading and localization of the optic disc and fovea center. The images in the dataset were acquired using a Kowa VX-10$\alpha$ digital camera with a $50^{\circ}$ field of view (FOV) and a resolution of 4288x2848 pixels in jpg file format. Annotation of the dataset was implemented by individuals with expertise, including a master's student, a PhD student, and a medical expert, with validation conducted by a retinal specialist. The dataset contains a total of 516 images, which have been split into a training set (413 images) and a testing set (103 images) for both DR/DME severity grading and ocular structure localization. However, for the lesion segmentation task, there are only 81 images.

- **Messidor (Decencière et al., 2014):** The dataset was created between 2006 and 2008, collecting 1200 retinal fundus images from three ophthalmology departments in France. These images were captured using a 3CCD color video camera mounted on a Topcon TRC NW6 non-mydriatic retinography device with a 45-degree field of view (FOV). The images have resolutions of 1440x960, 2240x1488, and 2304x1536 pixels. The dataset serves multiple purposes, including diabetic retinopathy (DR) severity grading, where labels range from 0 to 4; lesion segmenta-

tion, which was manually segmented by expert ophthalmologists for 30 percent of the dataset; and macula localization, where ground-truth locations were labeled for 1136 images.

- **Private datasets :** The datasets were collected from the ophthalmology departments of two hospitals in Thailand: Maharaj Nakhon Ratchasima Hospital and Suranaree University of Technology Hospital (SUTH). Each dataset was annotated into two classes: positive, indicating medically suitable retinal images, and negative, indicating medically unsuitable retinal images. The first dataset comprises 428 images of varying sizes, with 328 positive and 100 negative cases. The second dataset consists of 610 images with a resolution of 2976×2976 pixels, including 337 positive and 273 negative cases. Due to the small size of dataset, we use the cross-validation testing to test the algorithm's performance .

- **APTOS 2019 Blindness Detection (Karthik, 2019):** This dataset has been established with the aim of developing a medical screening solution for Aravind Eye Hospital in India, specifically to address the needs detect and prevent DR disease among numerous rural patients. The most effective solution identified will be distributed with other ophthalmologists through the 4th Asia Pacific Tele-Ophthalmology Society (APTOS) Symposium. The dataset comprises a total of 3,662 images in the training set, as illustrated in Figure 3.1 and 1,928 images in the testing set. These images exhibit various resolutions and are stored in png format, distributed across five distinct categories. The dataset is publicly accessible on Kaggle at the following link: (https://www.kaggle.com/competitions/aptos2019-blindness-detection).

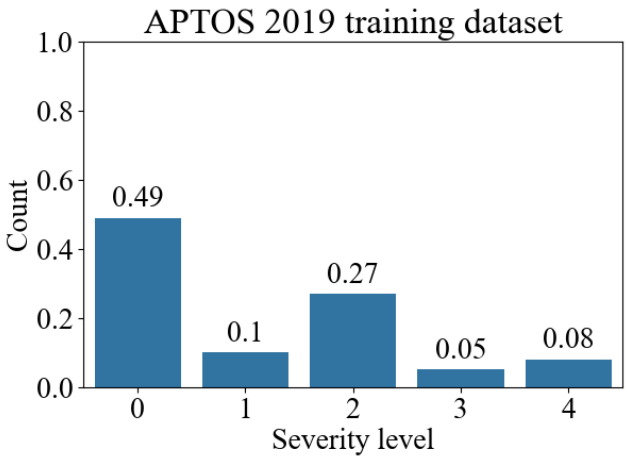


**Figure 3.1** The class distribution of these datasets reveals that class 0 (no diabetic retinopathy) is the majority class, indicating an imbalance issue. The total number of images in the training sets of APTOS 2019 is 3,662.

## 3.2 Image Screening

In this section, we implement 2 steps to achieve the screening of retinal images. The first step related to construct the templates for both the optic disc and macula. Then, we cooperate these templates and correlation filtering technique to detect the existence of optic disc and macula in retinal image. The second step is to screen the retinal image by using the screening algorithm to identify the medically suitable retinal image.

### 3.2.1 Optic disc and macula detection

Due to our screening algorithm relating to apply the correlation filtering technique for detect the existence of optic disc and macula in retinal image. Hence, we initially utilize the training set of the IDRiD dataset to generate reference templates (or mask) for both the optic disc and macula. Before generating the reference templates, we preprocess the fundus images by cropping the redundant dark areas surrounding the images. This preprocessing step is essential for enhancing image quality by eliminating background noise and improving overall consistency. Subsequently, all images are resized to a standardized resolution of 1280×1280 pixels, Figure 3.2.

These preprocessing procedures ensure uniformity in both image dimensions

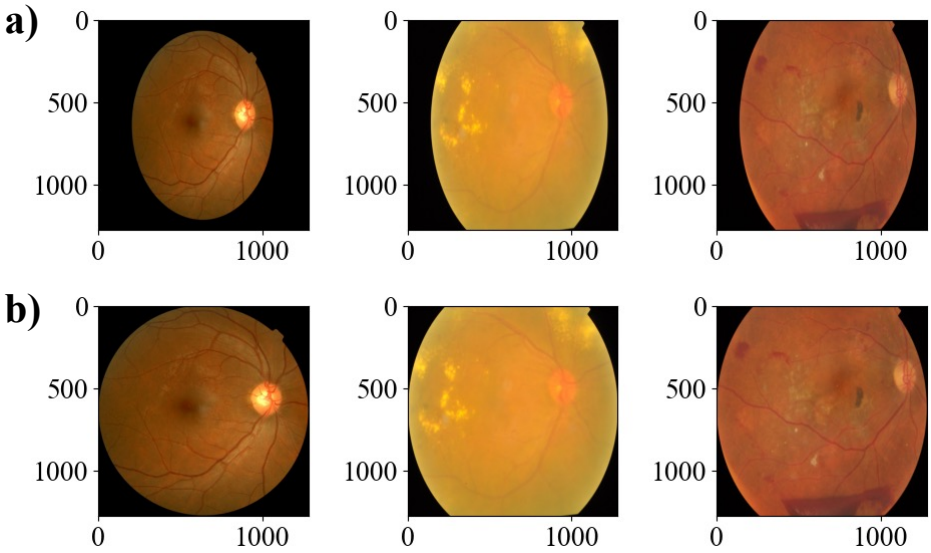


**Figure 3.2** the cropped dark field images, a) images without cropping, b) images with cropping.

and retinal region coverage, thereby supporting precise and consistent cropping during template generation. Then cropping the resized images around the optic disc and macula. For optic disc, the cropping size is varied in the range of 150 to 400 with the step size of 50 in both symmetric and asymmetric shape. For macula, the cropping size is varied in the range of 100 to 350 with the step size of 50 in both symmetric and asymmetric shape. Currently, we use the cropping dimensions of 300x350 and 200x300 for the optic disc and macula. This cropping procedure was repeated N times, and the resulting cropped images were averaged to create reference templates for the optic disc and macula, respectively, define as,

$$\bar{T}(x, y) = \frac{1}{N} \sum_{i=1}^{N} T_i(x, y) \tag{3.1}$$

where $\bar{T}$ is a reference template and $T_i$ is a cropped image number $i^{th}$. In the context of ocular detection, we deploy the correlation filtering method on the resized fundus image using an optic disc template to identify the optic disc.

Following this, we delineate the region of interest (ROI) for macula detection by utilizing the determined location of the optic disc as a central reference point and selecting an area that locate to threefold the size of the optic disc within a range of

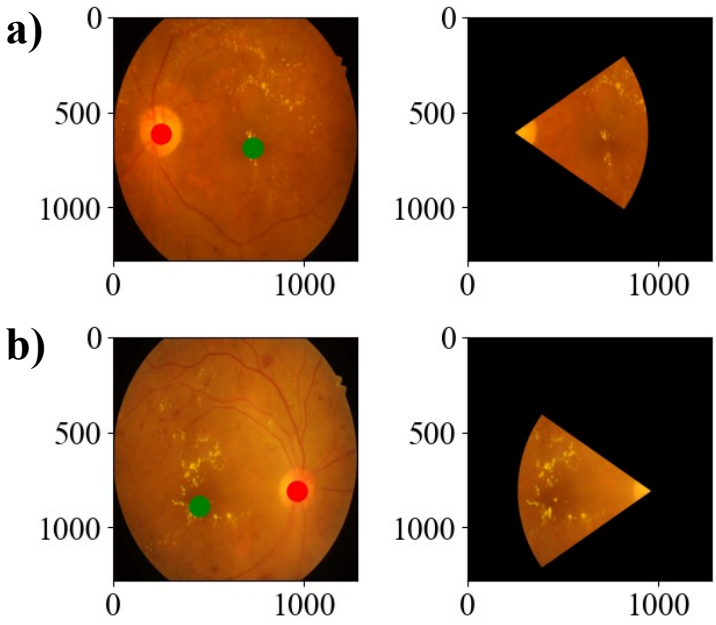


**Figure 3.3** The figure shows the result of ROI cropping from both eyes, a) the left eye and b) the right eye. The macula and optic disc's ground-truth locations are represented by the red and green dots.

-30 to 30 degrees (Sekhar et al., 2008), resulting in the ROI in a like-conical shape, illustrated in Figure 3.3. Subsequently, we employ the correlation filtering technique on the ROI image to locate the macula. Ultimately, thresholding the correspondence space is applied to determine the reliability of the maximum point, which serves as an indicator for locating the optic disc and macula. This process involves establishing a threshold value within the correspondence space, beyond which points are considered reliable. By applying thresholding, we can effectively identify the maximum point that accurately represents the location of these ocular structures. Furthermore, we currently use matching method, named normalized correlation coefficient (Eq. 3.2), to establish the correspondence space in the correlation filtering.

$$R(x,y) = \frac{\sum_{x',y'} \overline{T}'(x',y') \cdot I'(x+x',y+y')}{\sqrt{\sum_{x',y'} \overline{T}'(x',y')^2 \cdot \sum_{x',y'} I'(x+x',y+y')^2}} \tag{3.2}$$

where,

$$\bar{T}'(x', y') = \bar{T}(x', y') - \frac{1}{w \cdot h} \cdot \sum_{x'', y''} \bar{T}(x'', y'') \tag{3.3a}$$

$$I'(x + x', y + y') = I(x + x', y + y') - \frac{1}{w \cdot h} \cdot \sum_{x'', y''} I(x + x'', y + y'') \tag{3.3b}$$

Let I is a retinal fundus image.

### 3.2.2 Screening algorithm by rulebased

In this work, we use and develop a method proposed in (Şevik et al., 2014) to be suitable for our task in classifying the MURI and MSRI. We will call the retinal image MSRI if it can pass these criteria: first, the optic disc and macula must locate within the acceptance region, called R1; and second, the optic disc must locate outside the specific region, called R2. R2 is added to become a criterion because we don't need a nasal field of a retinal image. An example of these regions is illustrated in Figure 3.4. The R1 is defined by the lower and upper boundaries, following the below equations:

$$L_{R1} = y_c - \epsilon \tag{3.4a}$$

$$U_{R1} = y_c + \epsilon \tag{3.4b}$$

where $y_c$ is a vertical centerline of retinal image, $\epsilon$ is an adjusted parameter, $L_{R1}$ is a lower boundary, and $U_{R1}$ is an upper boundary. Moreover, the lower and upper boundary of R2 is computed as:

$$L_{R2} = y_c - \delta \tag{3.5a}$$

$$U_{R2} = y_c + \delta \tag{3.5b}$$

where $x_c$ is a horizontal centerline of retinal image, $\delta$ is an adjusted parameter, $L_{R2}$ is a lower boundary, and $U_{R2}$ is an upper boundary. In this work, we use grid search on 50 images of private dataset to determine the optimal parameters, thereby $\epsilon$ and $\delta$ is 180 and 220 pixels, respectively.

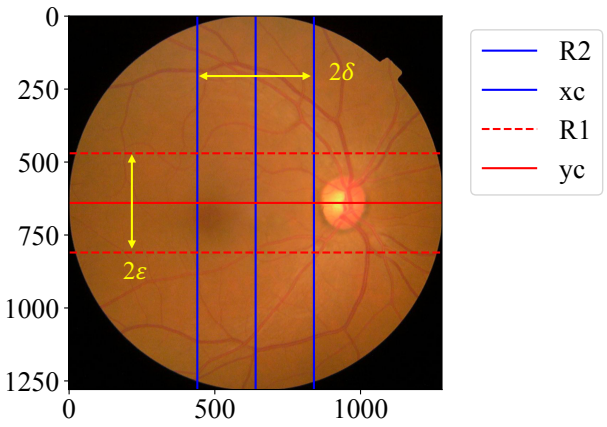


**Figure 3.4** The acceptance region of R1 and R2.

### 3.2.3 Screening algorithm by Machine learning (ML)

We further investigate the screening algorithm by integrating machine learning (ML) techniques. Upon successful ocular detection, tabular data are produced, comprising features extracted from retinal images, including $OD_{confidence}$, $M_{confidence}$, $OD_{position}$, and $M_{position}$. The confidence scores, ranging from 0 to 1, indicating the certainty of each detection. $OD_{position}$ and $M_{position}$ represent the coordinates of the detected optic disc and macula, respectively, in the [x, y] format. Furthermore, we derive additional features, including $Dy_{intercept}$ and $Dr_{distance}$, based on these primary features. $Dy_{intercept}$ denotes the vertical deviation between the linear regression line of $OD_{position}$–$M_{position}$ and a horizontal reference line positioned at the midpoint of the y-axis, evaluated at x=0, Figure 3.5. The $Dr_{distance}$ quantifies the displacement between $M_{position}$ and the center of the retinal image, measured by Euclidean distance, as shown in Figure 3.5. A representative example of the tabular data is provided in Table 3.1.

**Table 3.1** The tabular data features for ML model.

| gray!20 $OD_{confidence}$ | $M_{confidence}$ | $OD_{position}$ | $M_{position}$ | $Dy_{intercept}$ | $Dr_{distance}$ |
|---|---|---|---|---|---|
| 0.719 | 0.903 | [934, 808] | [535, 821] | 198.508 | 43577 |
| 0.835 | 0.774 | [261, 512] | [753, 589] | 168.848 | 15370 |
| 0.771 | 0.847 | [229, 560] | [456, 811] | 333.212 | 63097 |

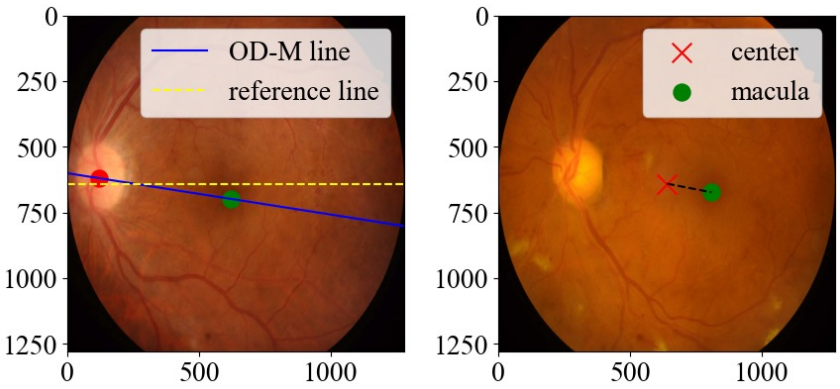


**Figure 3.5** The figure illustrates components of the image that are used to calculate the $Dy_{intercept}$ and $Dr_{distance}$ features. (left) The $Dy_{intercept}$ is the deviation between OD-M line and reference line on the y-axis at x = 0. (right) The $Dr_{distance}$ is the distance from the center of the image to the center of the macula.

We employ the tabular data extracted from the training set of our private dataset to train a variety of machine learning (ML) algorithms, including logistic regression, decision tree, support vector machine (SVM), random forest, histogram gradient boosting, light gradient boosting, and XG boosting. To optimize model performance prior to comparison, we apply grid search methods to each ML algorithm. The evaluation of each hyperparameter configuration is conducted using stratified 5-fold cross-validation. ML models, pipelines, and cross-validation procedures are mainly implemented using the Scikit-learn package (Pedregosa et al., 2011).

## 3.3 Evaluation of Image Screening

In the context of ocular detection, we evaluate the performance of our algorithm using the testing set of IDRiD. Initially, we quantify the accuracy of our algorithm's predictions by measuring the error between the predicted location and the ground-truth location of ocular structures. This analysis is conducted using the Euclidean distance (ED), providing insights into the precision of our predictions at the pixel level. Additionally, we generate the negative sample, which is a background image, to measure the algorithm's performance using other evaluation metrics, namely precision, recall, and AP score. This background image resembles the fundus image but lacks the optic disc and macula Figure 3.6. This step is crucial as it allows us to accurately determine

false positives (FP), which are difficult to define in fundus images containing all ocular structures. Therefore, in the quantitative measurement, we calculate the AP score to determine the algorithm's accuracy in bounding box prediction, higher AP score indicates better algorithm performance in object detection tasks. Moreover, we use a R-criterion score as mentioned in (Gegundez-Arias et al., 2013), to evaluate the performance of our macula detection on the Messidor dataset and due to the Messidor dataset comprising images of varying sizes, we apply different R values corresponding to each image size: R = 68 for images sized 1440 x 960, R = 103 for images sized 2240 x 1488, and R = 108 for images sized 2304 x 1536. In the context of screening, we employ various metrics, including the confusion matrix, false discovery rate (FDR), and recall of positive images, to assess the performance of our screening process on our private dataset. The false discovery rate is utilized to evaluate the proportion of negative samples that remain in the dataset after screening, while recall is used to measure the proportion of positive samples that are retained in the dataset post-screening.

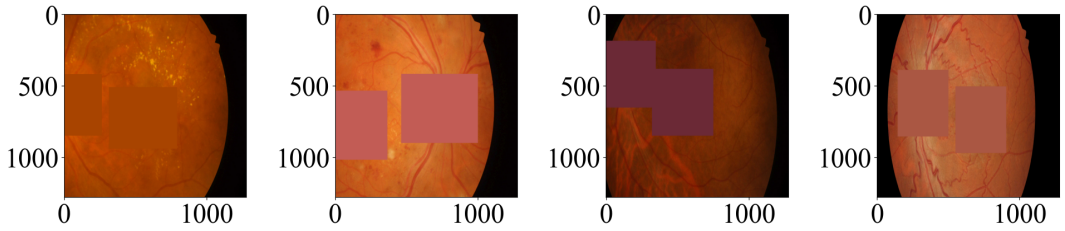


**Figure 3.6** The example of background images which generate by covering the optic disc and macula by the mean value of fundus image.

## 3.4 DR Grading

### 3.4.1 Data preparation

The APTOS 2019 dataset provide both training and testing set. Unfortunately, the testing set has no label, so we have to split the training set into training, validation and testing set with portion of 60%, 20% and 20%, respectively. Hence, the training set consists of 2,200 images, the validation set contains 731 images, and the testing set comprises 731 images. The APTOS 2019 dataset is highly imbalanced, with the distribution of severity levels as follows: 0 (No DR) - 1,083 images (49.2%), 1 (Mild DR) - 222 images (10.1%), 2 (Moderate DR) - 601 images (27.4%), 3 (Severe DR) - 117 images (5.3%), and 4 (Proliferative DR) - 177 images (8.0%).

### 3.4.2 Data augmentation and balancing

To address the imbalanced data issue, we implement oversampling technique, named Synthetic Minority Over-sampling Technique (SMOTE) (Chawla et al., 2002), to generate synthetic samples for the minority classes. However, our input data is not tabular data, but rather images. Therefore, we flatten the image into a vector with the size of HxWxC, where H is image height; W is image width; and C is number of image channel, and then, apply the SMOTE technique to generate new samples through interpolation between existing samples, as shown in Figure 3.7. Additionally, we also apply data augmentation techniques to enhance the diversity of our training set. These techniques include random horizontal flipping, random color jitter, random Gaussian blur, random adjust sharpness, random auto contrast, and random cropping. The augmentation process is performed on-the-fly during training to ensure that the model encounters a wide range of variations in the input data.

### 3.4.3 Architecture

Our grading network comprises of two modules: a backbone model for feature extractor and a custom FCNN for classifier. The feature extractor is responsible for extracting high-level features from the input fundus images, and then, classifier leverages these features to predict the serverity level of diabetic retinopathy (DR). Our backbone model is Swin Transformer, which is a type of vision transformer that has shown promising results in various computer vision tasks. The classifier is FCNN, which contain

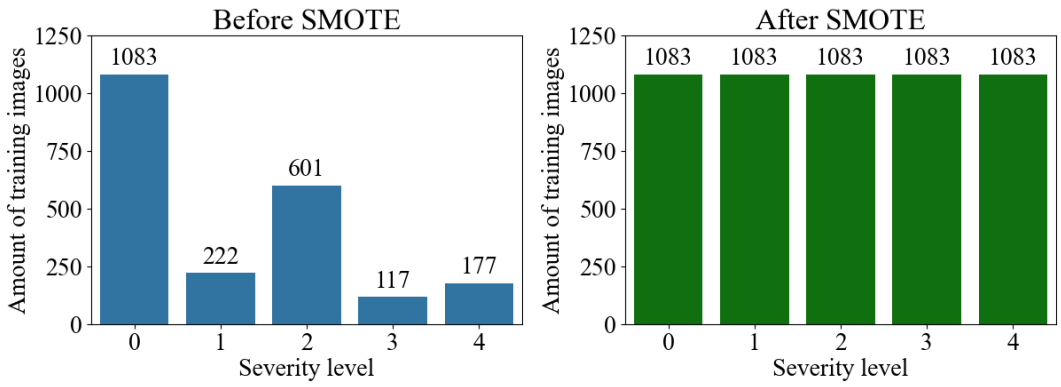


**Figure 3.7** The distribution of the synthetic samples generated by SMOTE in the prior and posterior stage.

5 layers and number of node in each layer is 256, 128, 128, 64, and 5, respectively. Moreover, the essential detail of Swin Transformer is described in below.

### Swin Transformer

The Swin Transformer (Liu et al., 2021) is a hierarchical vision transformer that is designed to address the challenge of processing high-resolution images efficiently in ViT-base model because Transformers is designed for language task, thereby adapting to vision task lead to the issue of computaional complexity, that is quadratic growing corresponding to image size. Therefore, the Swin Transformer introduces a Window-based self-attention mechanism, which partition the input image into non-overlapping windows and applies self-attention within each window, resulting in reformulating the complexity time to be linear, while the standard is quadratic complexity as a function of number of patch. As a result of this windowing scheme, Swin Transformer can understand the local context of image. Thus, to compromise the global context, Shifted Window-based self-attention is introduced, which shifts the windows between consecutive patch, allowing the model to capture global context in the image. This approach significantly reduces the computational cost while maintaining the ability to capture long-range dependencies in the image. Moreover, the Swin Transformer architecture is characterized by its hierarchical structure, where the feature maps are progressively downsampled, allowing the model to learn multi-scale representations of the input image similar to CNN 3.8. Eventually, this design enables the Swin Transformer to balance efficiency and complexity resulting in state-of-the-art performance across a various

fields of computer vision tasks, including image classification, object detection, and semantic segmentation.

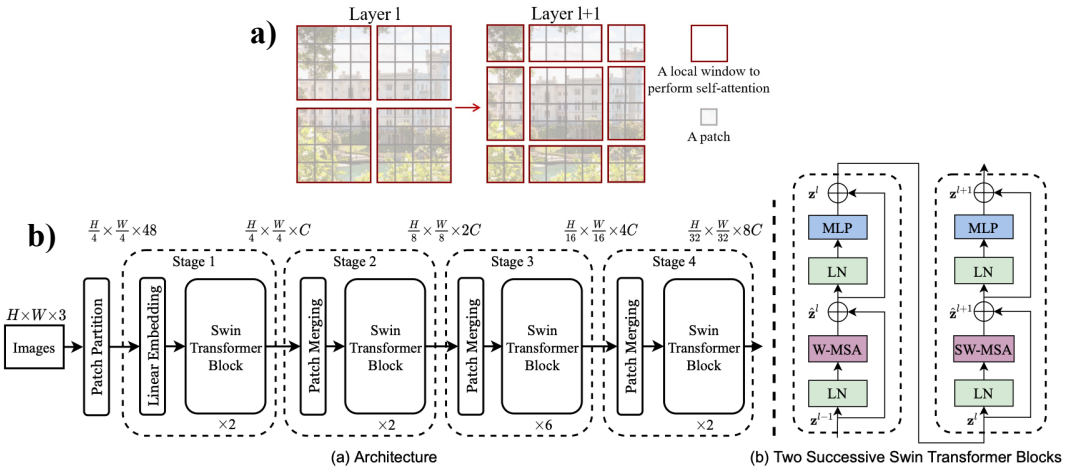


**Figure 3.8** The architecture and attention mechanism of the Swin Transformer. a) The window shifting mechanism allow the model to capture global and nearest neihbour context. b) Swin Transformer architecture and blocks

### 3.4.4 Training setting and strategy

The training images used are RGB images with a resolution of 512x512 pixels and a batch size of 8. Model optimization is performed using the cross-entropy loss function and the AdamW optimizer with default parameters. To improve model robustness and generalization, various data augmentation techniques are employed during training. Model fine-tuning is conducted using a slow unfreezing strategy, in which the pretrained model's layers are gradually unfrozen and trained for a limited number of epochs to allow adaptation to the target task while preserving the learned representations. Initially, the classifier head is trained while the backbone remains frozen. Subsequently, the training proceeds with a gradual unfreezing of the backbone layers, starting from the last 20 layers of the Swin Transformer, unfrozen and trained for 50 epochs. The learning rate is set to 1e-3, and a step learning rate scheduler is employed with a step size of 10 and a decay factor (gamma) of 0.1. Model checkpoints are saved every 5 epochs, and the lowest validation loss checkpoint is retained to mitigate overfitting. Early stopping is not applied, as the entire 50-epoch training progression is analyzed. Subsequently, the number of unfrozen layers is incrementally increased

to 70, 160, and 250, respectively, with each stage undergoing an additional 50 epochs of training. This results in a total of 250 training epochs. The unfreezing schedule is designed to incrementally increase the number of trainable layers by the percentage of backbone parameters including 20, 40, 60, and 80 percents.

## 3.5 Evaluation of DR Grading

Deep learning has frequently demonstrated remarkable performance across a multitude of tasks. However, leveraging deep learning without meticulous evaluation can be likened to deploying an unaccredited ophthalmologist, trained but lacking certification to assess performance, thereby compromising the reliability of diagnoses. Therefore, employing concise evaluation metrics is imperative to increase the reliability and confidence in the predictions generated by our network. Primarily, we measure the prediction performance by using the multi-class confusion matrix, which provides insights into the number of correct and missed predictions. As a result of this matrix, we can compute metrics such as precision, recall, and F1 score for each class, thereby offering a comprehensive evaluation of the network's classification capabilities. Notably, all evaluation results are reported using three significant figures, corresponding to the scale of the smallest amount of classes used in this study, which contains approximately 100 samples. This level of precision ensures consistency with the data resolution. Furthermore, we also demonstrate the network performance on the threshold-independent metric, including the ROC curve and AUC. Finally, we utilize the QWK metric to evaluate the grading performance of the network, providing a holistic assessment of its efficacy in classifying DR severity levels.

## 3.6 Computational Resources

The experiments in this work were conducted on a personal computer equipped with an NVIDIA RTX 3070 Ti GPU with 8 GB of VRAM, an Intel Core i7-13700K CPU, and 32 GB of RAM. The operating system used is Ubuntu 20.04 LTS, and the deep learning framework employed is PyTorch version 2.6 (Paszke, 2019).