Foundation Models in Retinal Images

RETFOUND

Article

A foundation model for generalizable disease detection from retinal images

https://doi.org/10.1038/s41586-023-06555-x

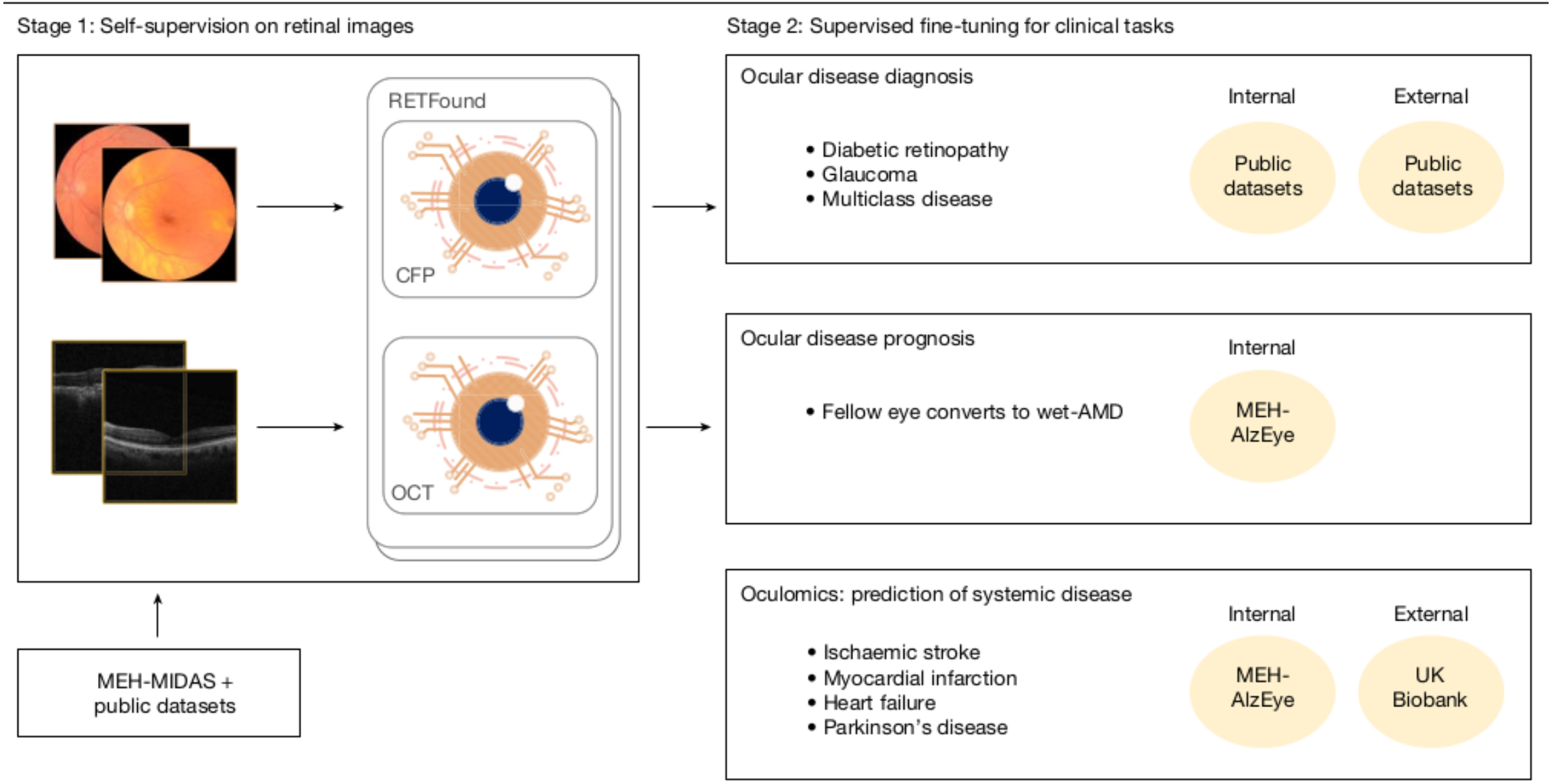
Received: 5 December 2022

Accepted: 18 August 2023

Published online: 13 September 2023

Yukun Zhou[1,2,3], Mark A. Chia[2,4], Siegfried K. Wagner[2,4], Murat S. Ayhan[1,2,4], Dominic J. Williamson[1,2,4], Robbert R. Struyven[1,2,4], Timing Liu[2], Moucheng Xu[1,3], Mateo G. Lozano[2,5], Peter Woodward-Court[1,2,6], Yuka Kihara[7,8], UK Biobank Eye & Vision Consortium*, Andre Altmann[1,3], Aaron Y. Lee[7,8], Eric J. Topol[9], Alastair K. Denniston[10,11], Daniel C. Alexander[1,12] & Pearse A. Keane[2,4]

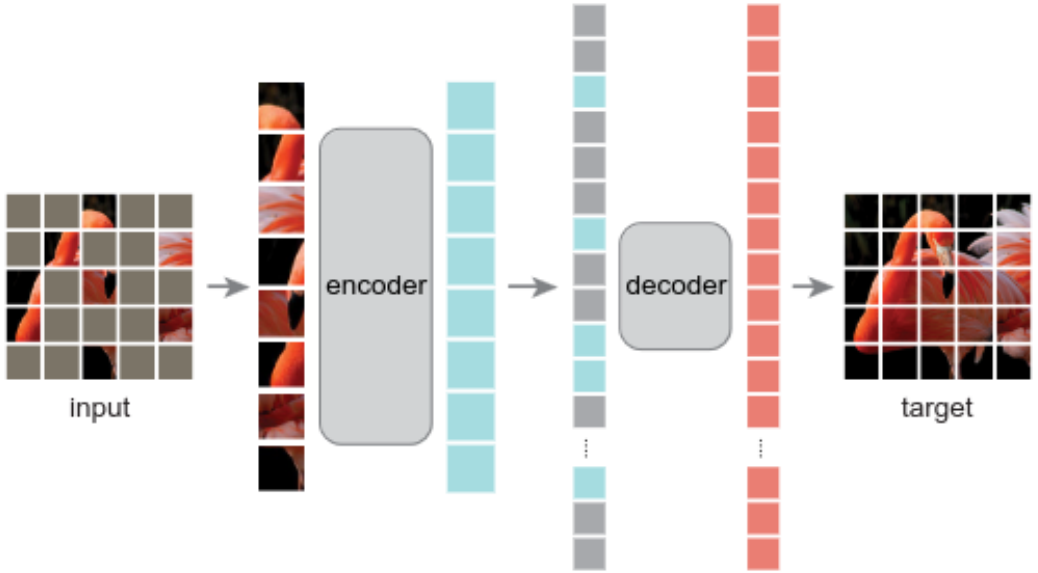
- Model trained using unlabelled images using self-supervised learning
- Pretrained on 1.6 million images - color fundus photography (CFP) and Optical Coherence Tomography (OCT)
- Dataset: MEH-MIDAS+Kaggle EyePacs (CFPs) + Kermany Dataset (OCT)
  - CFP's - 904,170
  - OCT's - 736,44
- Task-specific fine-tuning - diagnosis, prognosis, prediction of systemic disorders
- Masked Autoencoder for SSL.
- Also experiment with SimCLR, SwAV, DINO
- First pre-trained on ImageNet and then on Retinal Images


Stage 1: Self-supervision on retinal images
RETFound
CFP
OCT
MEH-MIDAS + public datasets
Stage 2: Supervised fine-tuning for clinical tasks
Ocular disease diagnosis
• Diabetic retinopathy
• Glaucoma
• Multiclass disease
Internal
Public datasets
External
Public datasets
Ocular disease prognosis
• Fellow eye converts to wet-AMD
Internal
MEH-AlzEye
Oculomics: prediction of systemic disease
• Ischaemic stroke
• Myocardial infarction
• Heart failure
• Parkinson's disease
Internal
MEH-AlzEye
External
UK Biobank

Preprocessing and Augmentations

- AutoMorph
- Resized to 256x256
- OCT - middle portion extracted
- Random cropping and resizing to 224x224,
- Random Horizontal Flipping
- image normalization

Masked Autoencoder



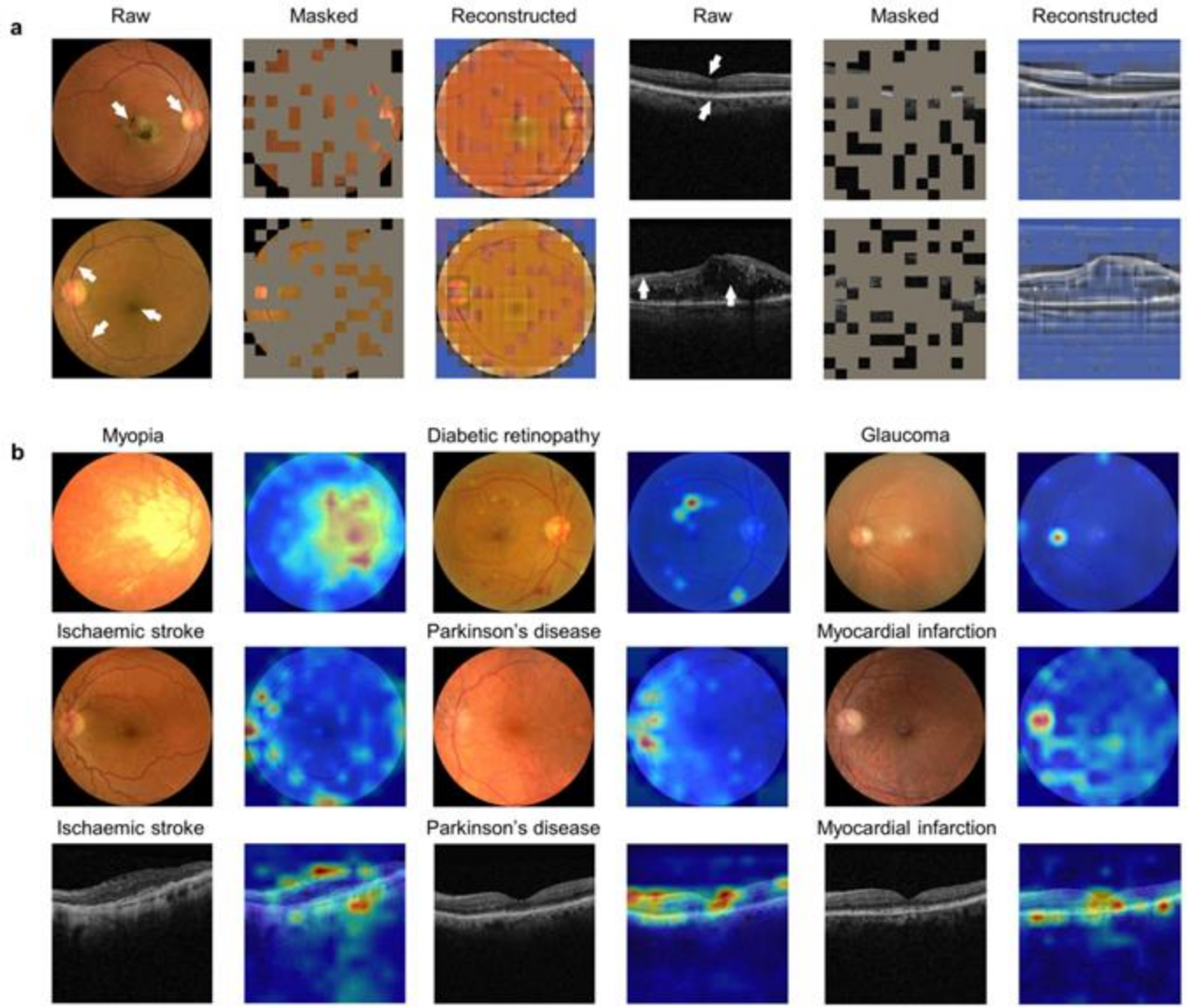
- Encoder : VIT-Large : 24 Layers, Embedding Size : 1024
- Decoder: VIT Small : 8 layers, Embedding Size: 512

Architecture:

- 16x16 patches are extracted from the image
- Linear Embedding + Position Embedding
- Randomly Sample Patches and Mask them
- Unmasked normal patches fed to encoder
- Learnable Mask Tokens with output of encoder is fed to decoder (position embedding)
- Output is reshaped to produce target output
- MSE Loss is used as Error Function

He et.al -Masked Autoencoders Are Scalable Vision Learners

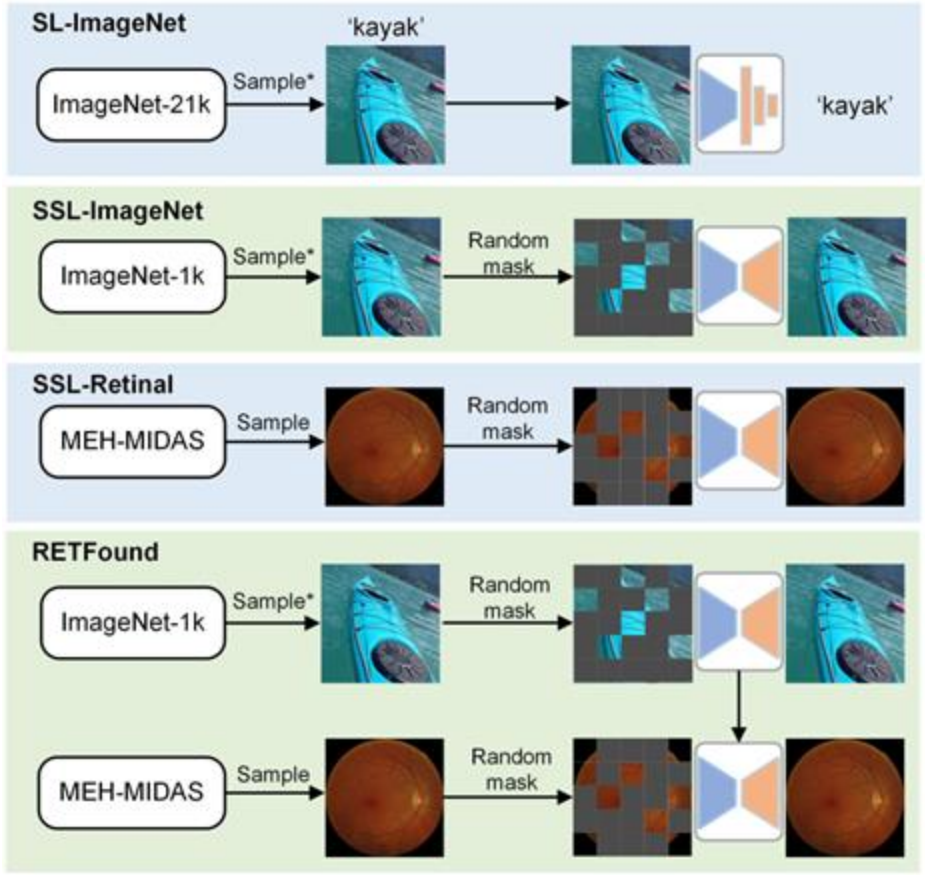
Qualitative Analysis on Retinal Images

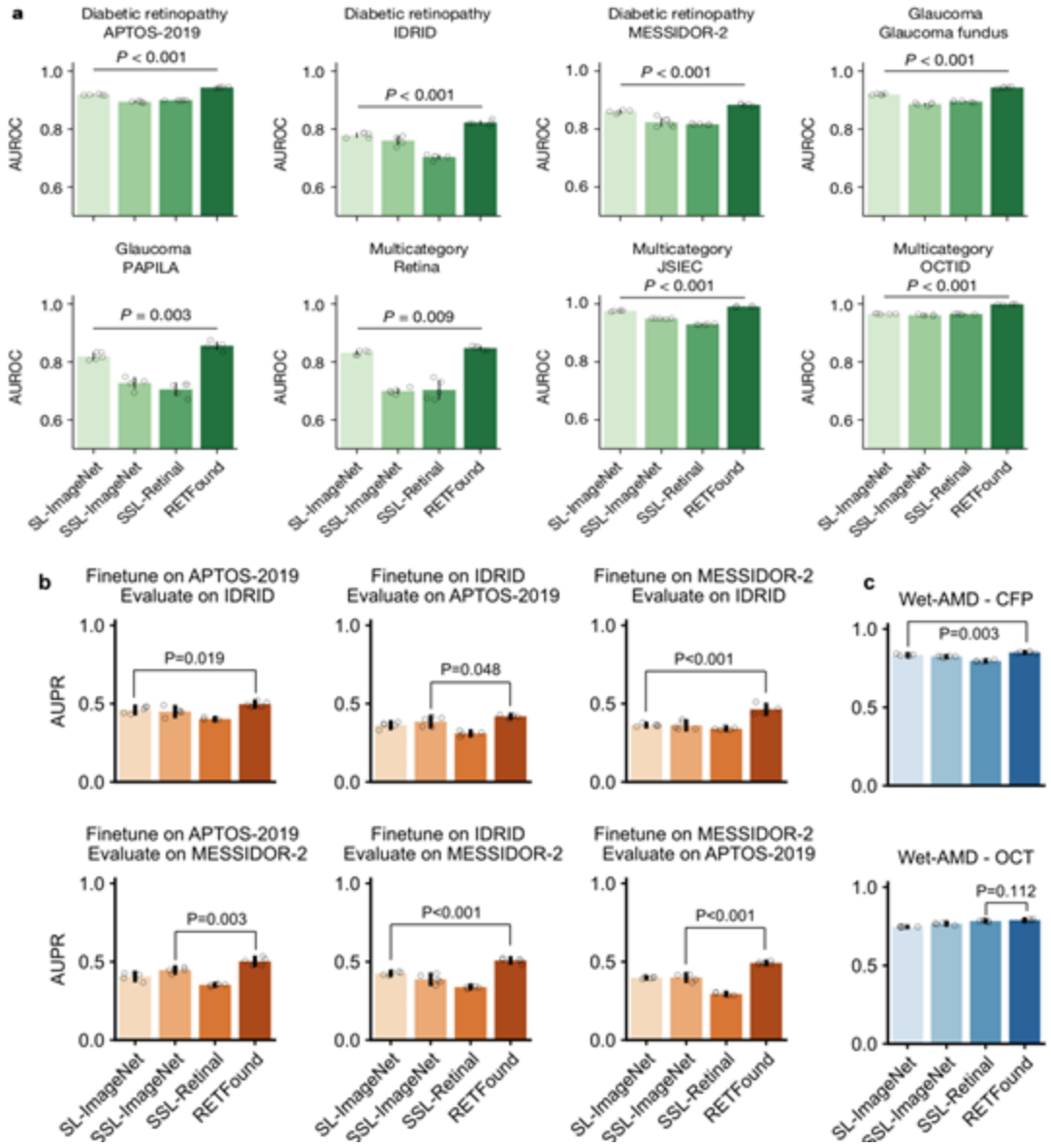


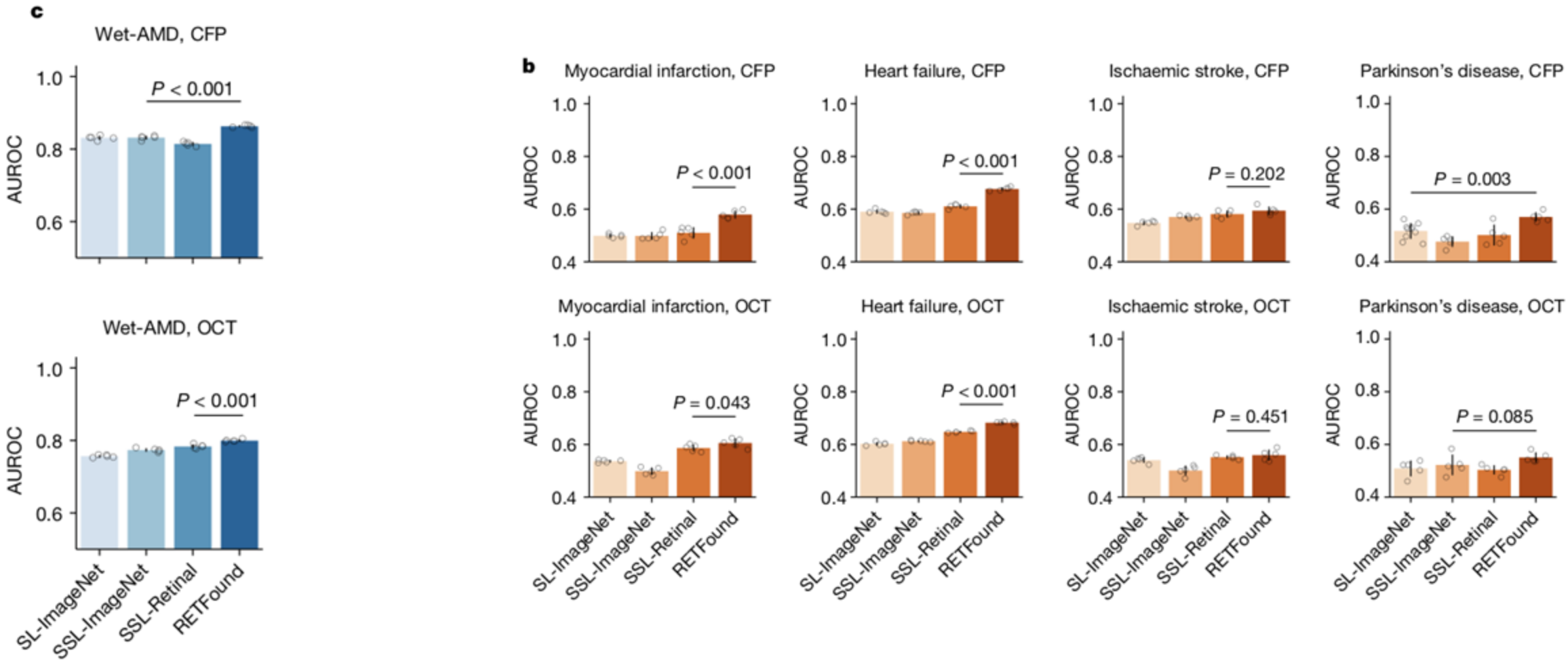
Downstream task

- Only Encoder is used for downstream tasks
- Tested for
  - Ocular Disease Diagnosis
  - Ocular Disease Prognosis
  - Systemic Disease Prediction

Comparison with other Approaches

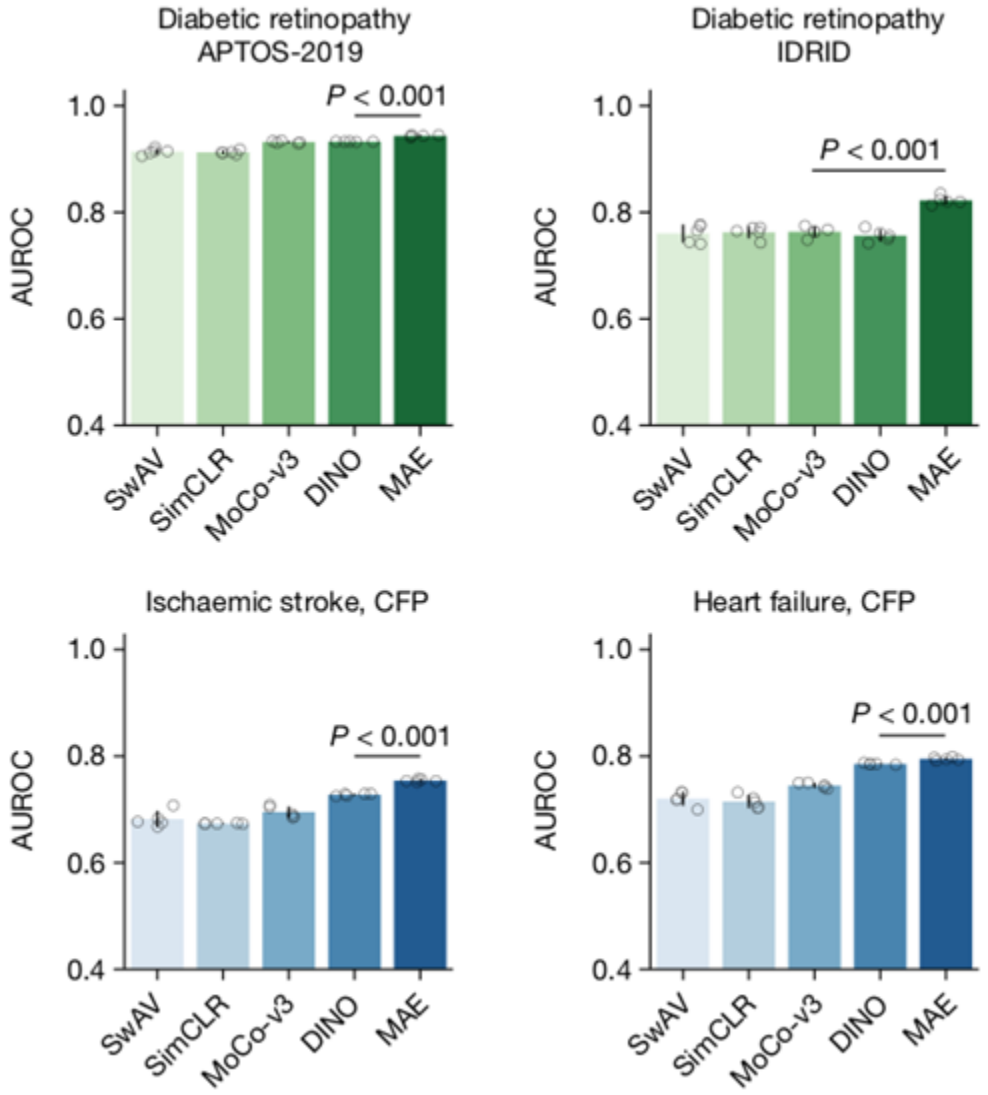




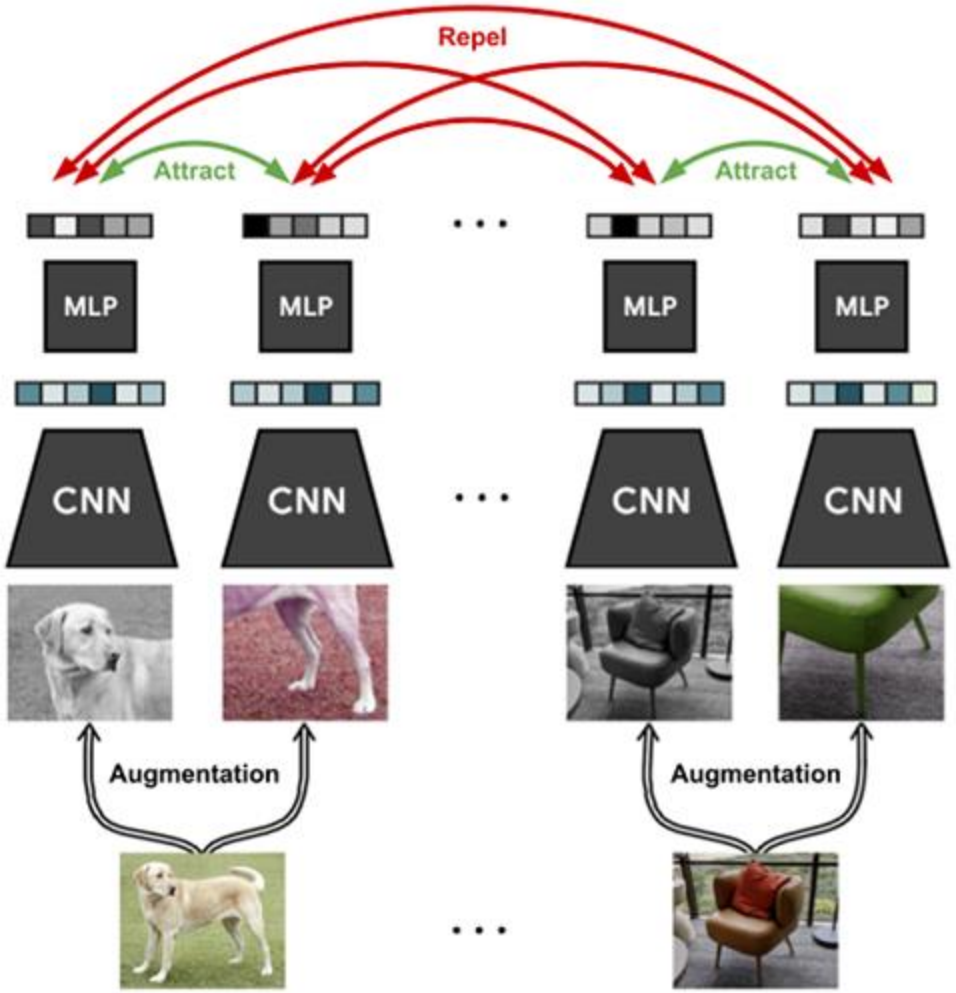

c
Wet-AMD, CFP
P < 0.001
AUROC
1.0
0.8
0.6
Wet-AMD, OCT
P < 0.001
AUROC
1.0
0.8
0.6
SL-ImageNet
SSL-ImageNet
SSL-Retinal
RETFound
b
Myocardial infarction, CFP
P < 0.001
Heart failure, CFP
P < 0.001
Ischaemic stroke, CFP
P = 0.202
Parkinson's disease, CFP
P = 0.003
Myocardial infarction, OCT
P = 0.043
Heart failure, OCT
P < 0.001
Ischaemic stroke, OCT
P = 0.451
Parkinson's disease, OCT
P = 0.085
AUROC
1.0
0.8
0.6
0.4
SL-ImageNet
SSL-ImageNet
SSL-Retinal
RETFound

Comparison of different SSL approaches

- MAE
- DINO
- MoCo - V3
- SimCLR
- Swav



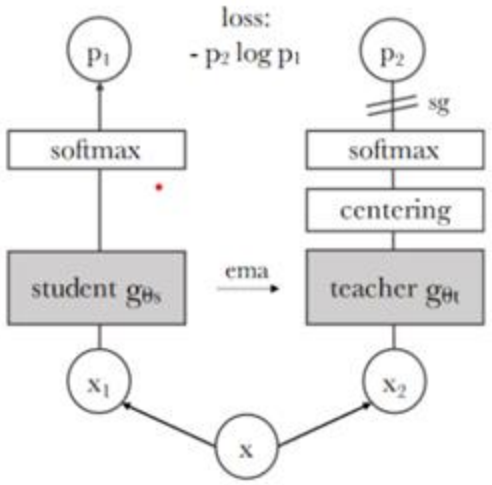
SimCLR



Algorithm 1 SimCLR's main learning algorithm.

input: batch size N, constant τ, structure of $f, g, \mathcal{T}$.
for sampled minibatch $\{\boldsymbol{x}_k\}_{k=1}^{N}$ **do**
  for all $k \in \{1, \ldots, N\}$ **do**
    draw two augmentation functions $t \sim \mathcal{T}, t' \sim \mathcal{T}$
    # the first augmentation
    $\tilde{\boldsymbol{x}}_{2k-1} = t(\boldsymbol{x}_k)$
    $\boldsymbol{h}_{2k-1} = f(\tilde{\boldsymbol{x}}_{2k-1})$ # representation
    $\boldsymbol{z}_{2k-1} = g(\boldsymbol{h}_{2k-1})$ # projection
    # the second augmentation
    $\tilde{\boldsymbol{x}}_{2k} = t'(\boldsymbol{x}_k)$
    $\boldsymbol{h}_{2k} = f(\tilde{\boldsymbol{x}}_{2k})$ # representation
    $\boldsymbol{z}_{2k} = g(\boldsymbol{h}_{2k})$ # projection
  end for
  for all $i \in \{1, \ldots, 2N\}$ and $j \in \{1, \ldots, 2N\}$ **do**
    $s_{i,j} = \boldsymbol{z}_i^\top \boldsymbol{z}_j / (\|\boldsymbol{z}_i\| \|\boldsymbol{z}_j\|)$ # pairwise similarity
  end for
  define $\ell(i, j)$ **as** $\ell(i, j) = -\log \frac{\exp(s_{i,j}/\tau)}{\sum_{k=1}^{2N} \mathbb{1}_{[k \neq i]} \exp(s_{i,k}/\tau)}$
  $\mathcal{L} = \frac{1}{2N} \sum_{k=1}^{N} [\ell(2k-1, 2k) + \ell(2k, 2k-1)]$
  update networks f and g to minimize $\mathcal{L}$
end for
return encoder network $f(\cdot)$, and throw away $g(\cdot)$

DINO



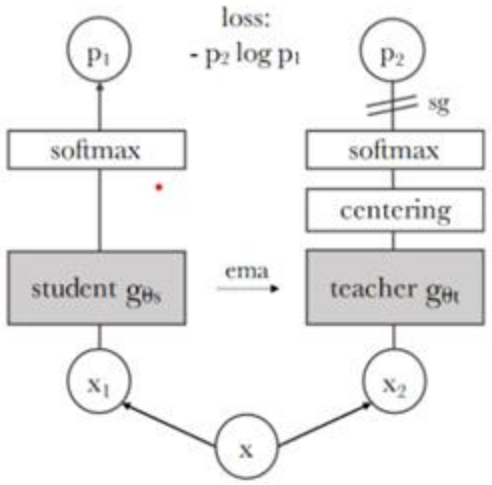
Self **Di**stillation with **No** Labels

$$P_s(x)^i = \frac{\exp(g_s(x)^i/T_s)}{\sum_{k=1}^{K}\exp(g_s(x)^k/T_s)}$$

$$P_t(x)^i = \frac{\exp(g_t(x)^i/T_t)}{\sum_{k=1}^{K}\exp(g_t(x)^k/T_t)}$$

$$\min_{\theta_s} H(P_t(x), P_s(x))$$

DINO



Self **Di**stillation with **No** Labels

$$P_s(x)^i = \frac{\exp(g_s(x)^i/T_s)}{\sum_{k=1}^{K}\exp(g_s(x)^k/T_s)}$$

$$P_t(x)^i = \frac{\exp(g_t(x)^i/T_t)}{\sum_{k=1}^{K}\exp(g_t(x)^k/T_t)}$$

$$\min_{\theta_s} H(P_t(x), P_s(x))$$

Multi Crop Strategy

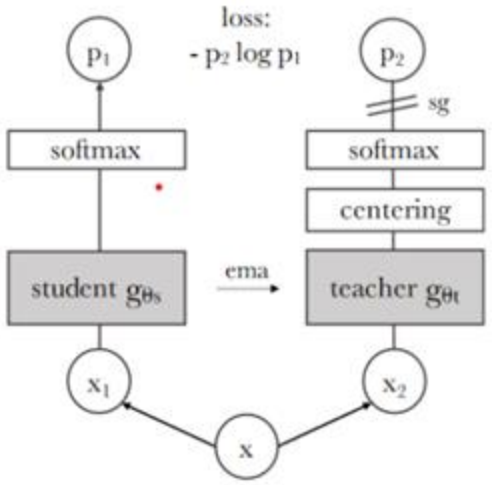
Global Views

Local Views

- The student processes all crops, whereas only the teacher processes the global views
- Minimise aggregated cross-entropy between all pairs of augmented views

$$\min_{\theta_s} \sum_{x\in\{x_1^g, x_2^g\}} \sum_{x'\in V, x'\neq x} H(P_t(x), P_s(x')).$$

DINO



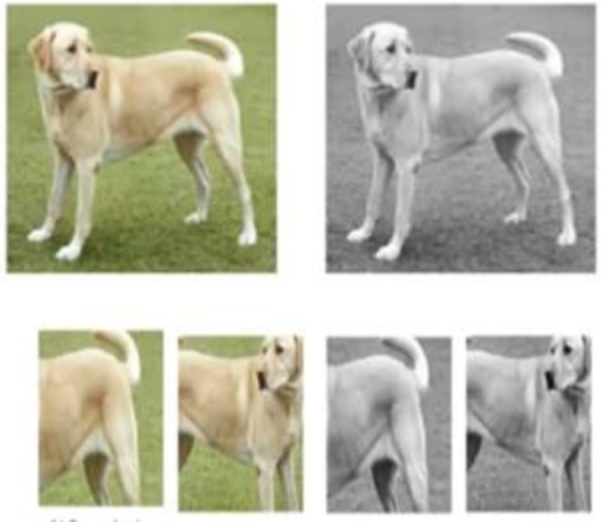
Self **Di**stillation with **No** Labels

$$P_s(x)^i = \frac{\exp(g_s(x)^i/T_s)}{\sum_{k=1}^{K}\exp(g_s(x)^k/T_s)}$$

$$P_t(x)^i = \frac{\exp(g_t(x)^i/T_t)}{\sum_{k=1}^{K}\exp(g_t(x)^k/T_t)}$$

$$\min_{\theta_s} H(P_t(x), P_s(x))$$

Multi Crop Strategy



Global Views

Local Views

- The student processes all crops, whereas only the teacher processes the global views
- Minimise aggregated cross-entropy between all pairs of augmented views

$$\min_{\theta_s} \sum_{x\in\{x_1^g, x_2^g\}} \sum_{x'\in V, x'\neq x} H(P_t(x), P_s(x')).$$

Both the networks use the same architecture.They have a backbone (ViT or ResNet) and a MLP projection head.

- Fixed Teacher Network while student is learning
- Update Teacher using EWMA

$$\theta_t = \lambda\theta_t + (1-\lambda)\theta_s$$

Centering Update

$$c \leftarrow mc + (1-m)\frac{1}{B}\sum_{i=1}^{B} g_{\theta_t}(x_i),$$

DINO ALGORITHM

Algorithm 1 DINO PyTorch pseudocode w/o multi-crop.

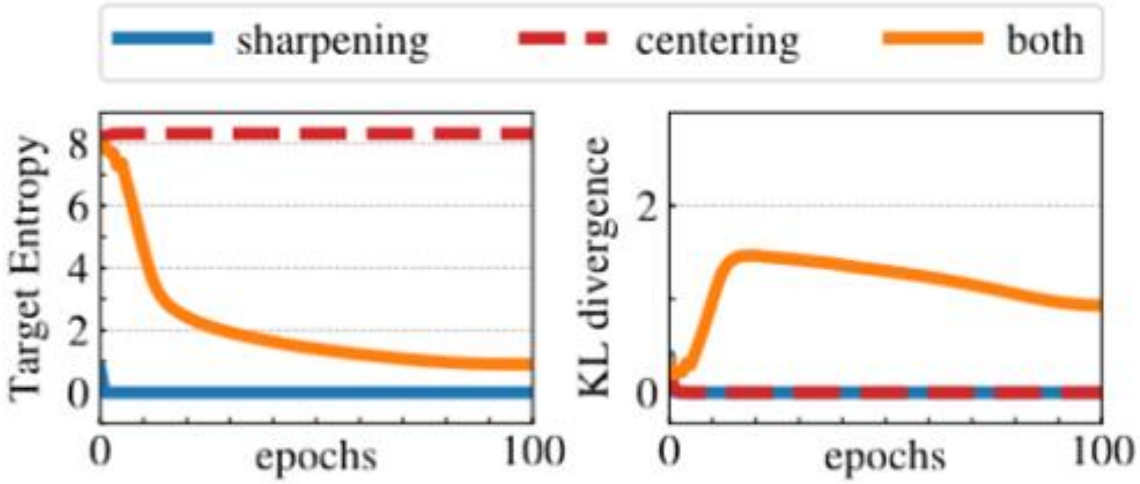
```
# gs, gt: student and teacher networks
# C: center (K)
# tps, tpt: student and teacher temperatures
# l, m: network and center momentum rates
gt.params = gs.params
for x in loader: # load a minibatch x with n samples
    x1, x2 = augment(x), augment(x) # random views

    s1, s2 = gs(x1), gs(x2) # student output n-by-K
    t1, t2 = gt(x1), gt(x2) # teacher output n-by-K

    loss = H(t1, s2)/2 + H(t2, s1)/2
    loss.backward() # back-propagate

    # student, teacher and center updates
    update(gs) # SGD
    gt.params = l*gt.params + (1-l)*gs.params
    C = m*C + (1-m)*cat([t1, t2]).mean(dim=0)

def H(t, s):
    t = t.detach() # stop gradient
    s = softmax(s / tps, dim=1)
    t = softmax((t - C) / tpt, dim=1) # center + shar
    return - (t * log(s)).sum(dim=1).mean()
```

DINO ALGORITHM

Algorithm 1 DINO PyTorch pseudocode w/o multi-crop.

```
# gs, gt: student and teacher networks
# C: center (K)
# tps, tpt: student and teacher temperatures
# l, m: network and center momentum rates
gt.params = gs.params
for x in loader: # load a minibatch x with n samples
    x1, x2 = augment(x), augment(x) # random views

    s1, s2 = gs(x1), gs(x2) # student output n-by-K
    t1, t2 = gt(x1), gt(x2) # teacher output n-by-K

    loss = H(t1, s2)/2 + H(t2, s1)/2
    loss.backward() # back-propagate

    # student, teacher and center updates
    update(gs) # SGD
    gt.params = l*gt.params + (1-l)*gs.params
    C = m*C + (1-m)*cat([t1, t2]).mean(dim=0)

def H(t, s):
    t = t.detach() # stop gradient
    s = softmax(s / tps, dim=1)
    t = softmax((t - C) / tpt, dim=1) # center + shar
    return - (t * log(s)).sum(dim=1).mean()
```

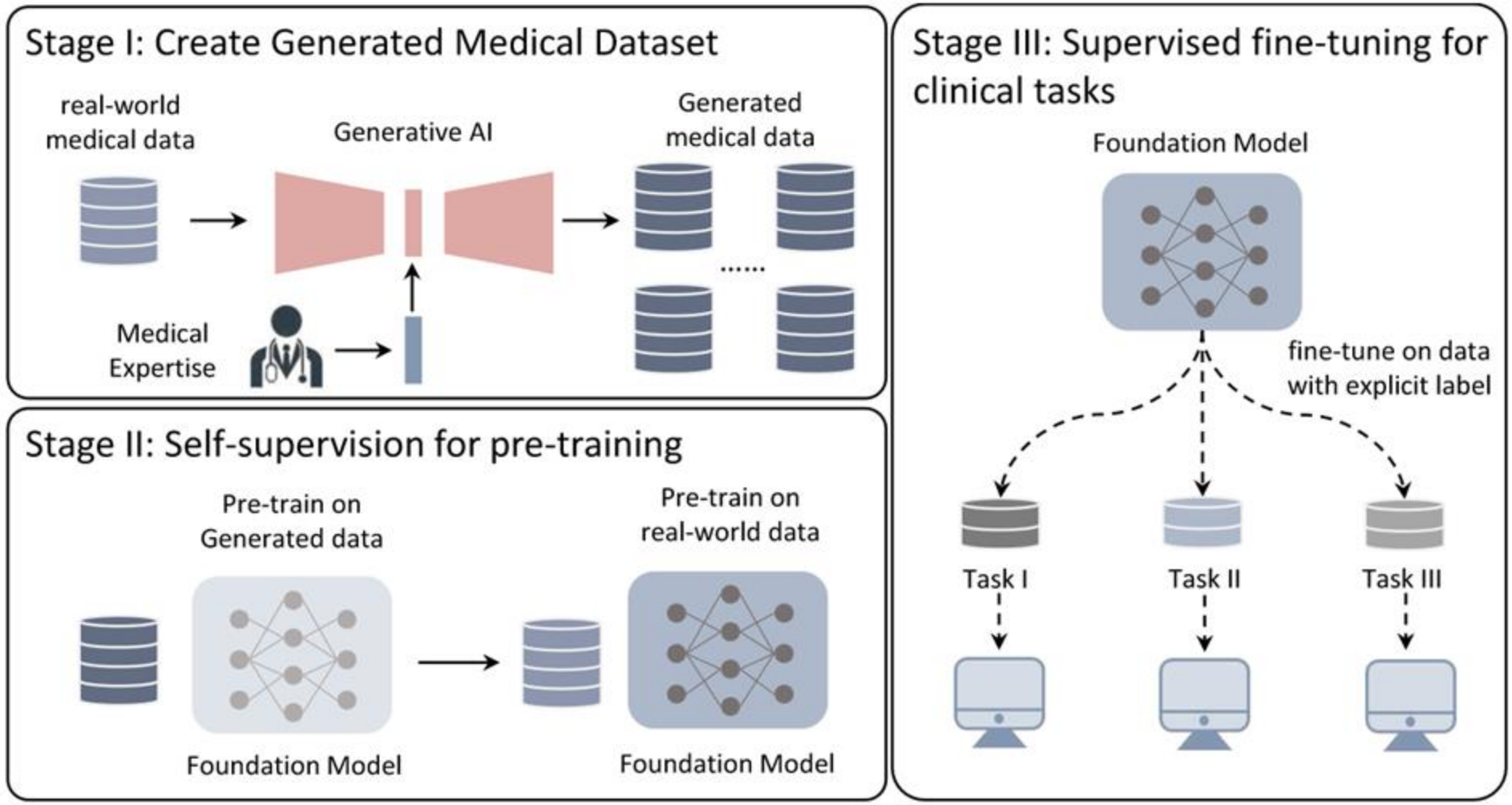
Avoiding Collapse



$$H(P_t, P_s) = h(P_t) + D_{KL}(P_t \| P_s)$$

- 2 types of collapse are possible:
  - output is uniform, irrespective of the input.
  - output is dominated by one feature
- centering prevents one dimension to dominate but encourages collapse to the uniform distribution, while the sharpening has the opposite effect.

Data Efficient Retfound (DERETFOUND)



Details

- Trained on publically available CFP Images
- Use 150k original images to train a Latent Diffusion Model (Stable Diffusion) to generate 1 million images
- Trained successively on real and generated images
- 600 epochs on generated images and 200 on real images
- Pretraining same as RETFOUND
- Computation : 8 X A100 (80 GB)GPUS

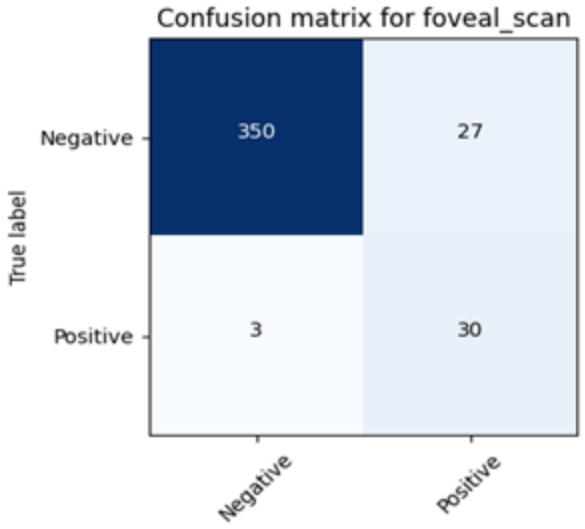
Dataset

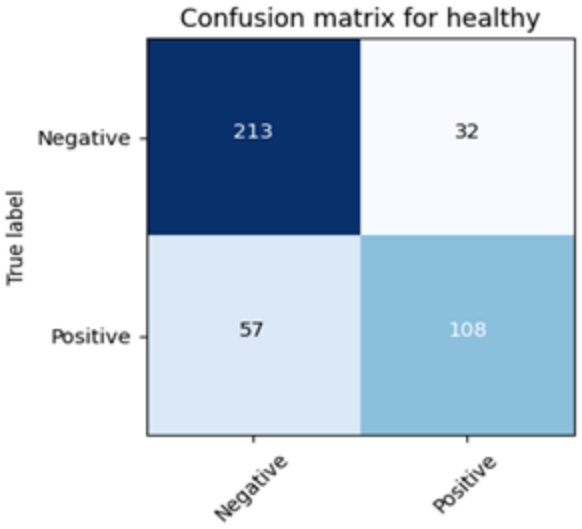
	TRAIN(1360)	TEST (410)
foveal scan	135	33
healthy	420	165
srf	89	2
irf	69	6
drusen	227	91
ped	220	27
hdots	840	257
hfoci	40	32

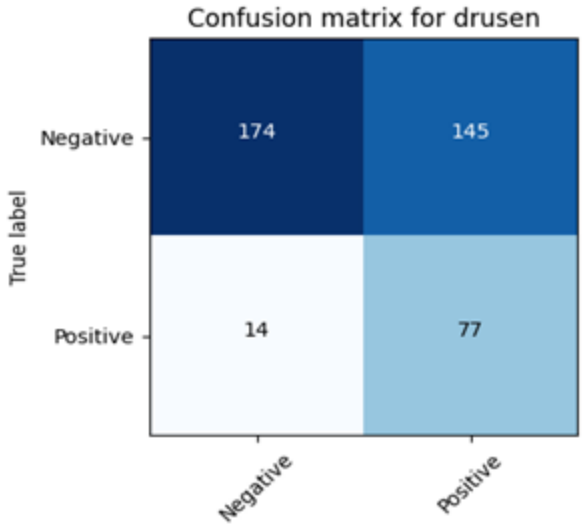
AU-ROC

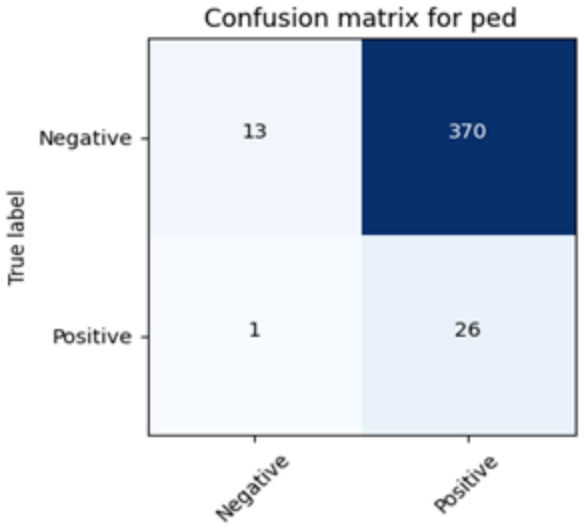
	Separate		Multitask	
	resnet50	retfound	resnet50	retfound
Healthy	0.83	0.8	0.8	0.81
Foveal Scan	0.94	0.95	0.96	0.91
Drusen	0.75	0.83	0.74	0.6
PED	0.74	0.76	0.69	0.74
hdots	0.64	0.66	0.64	0.67
Average	0.78	0.8	0.766	0.746

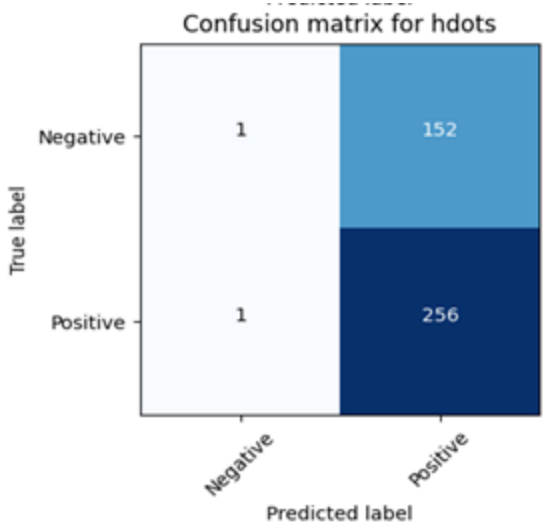
Common RESNET - Confusion Matrices



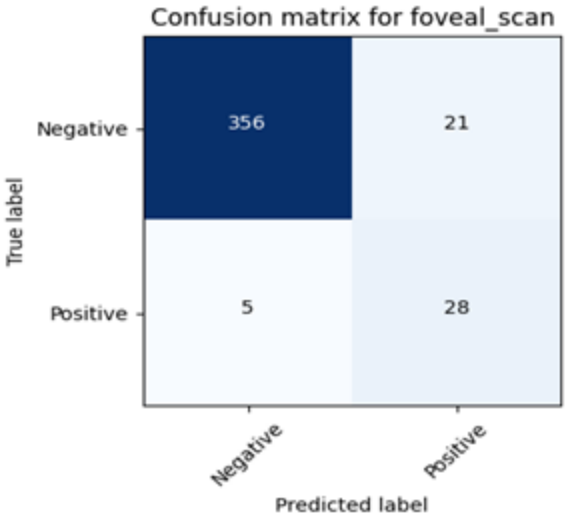


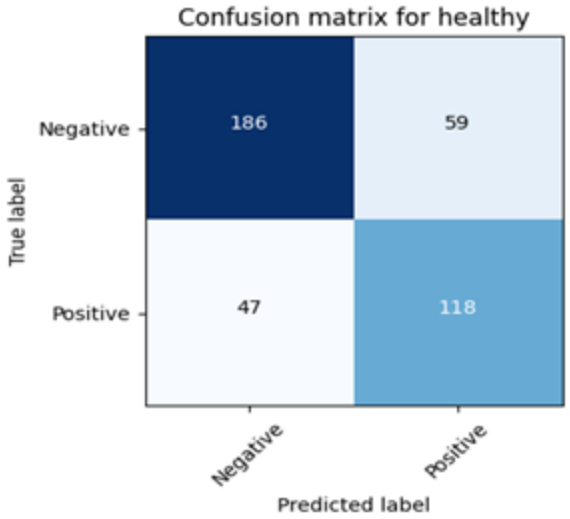


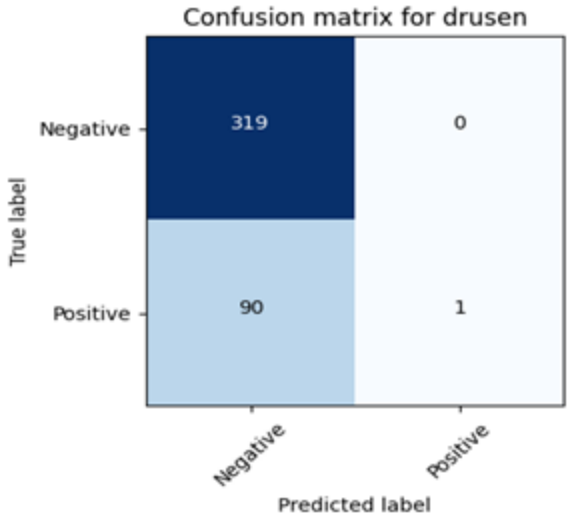


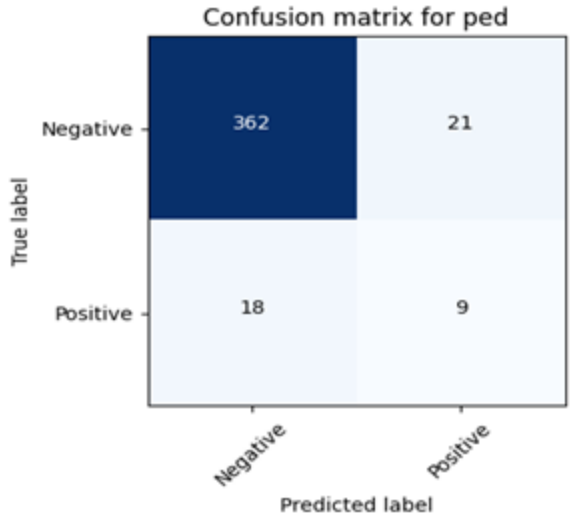


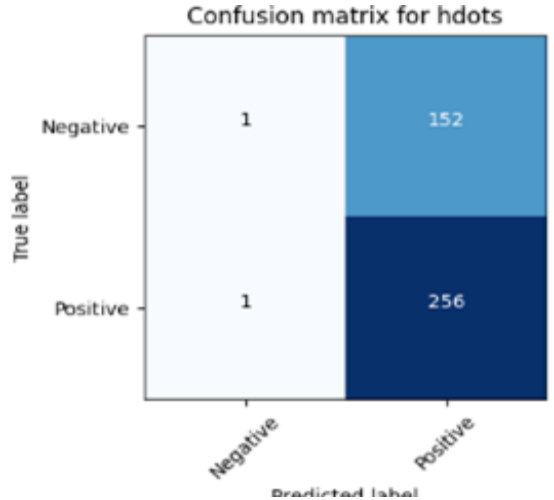
Common RETFOUND - Confusion Matrices



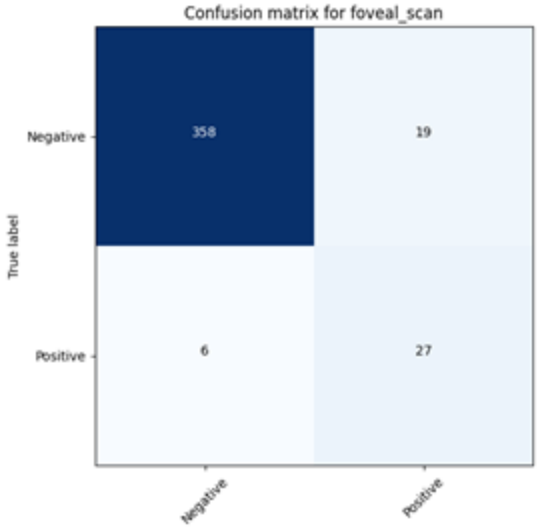


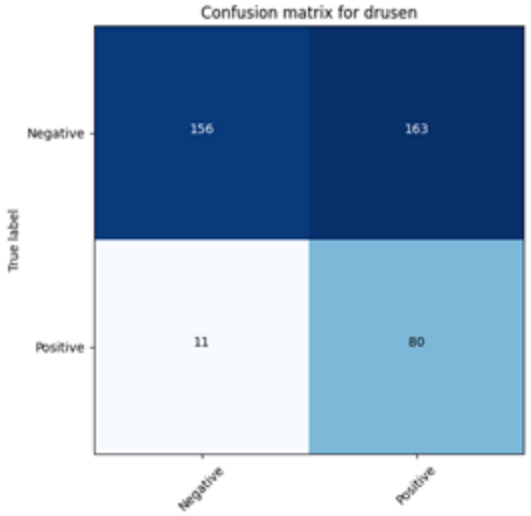


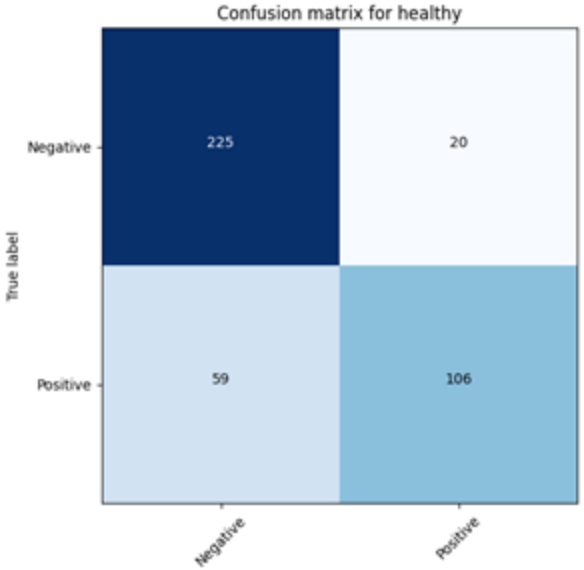


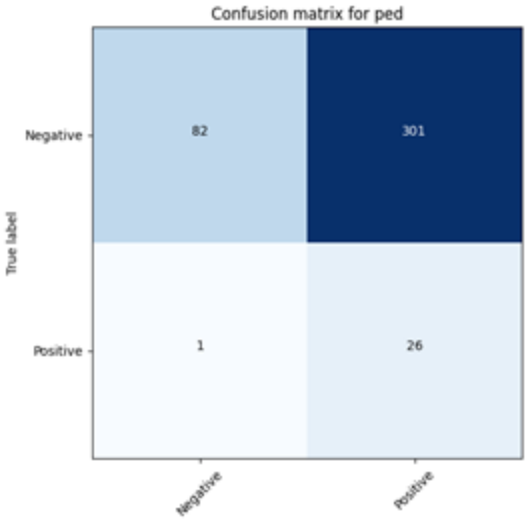


Separate RESNET - Confusion Matrices

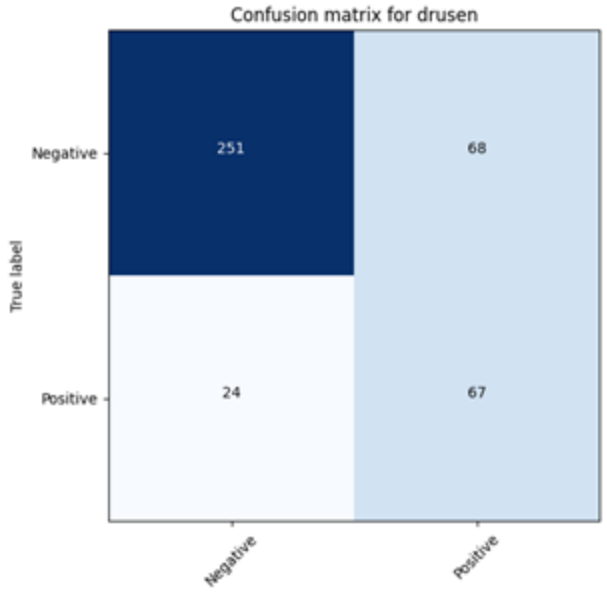


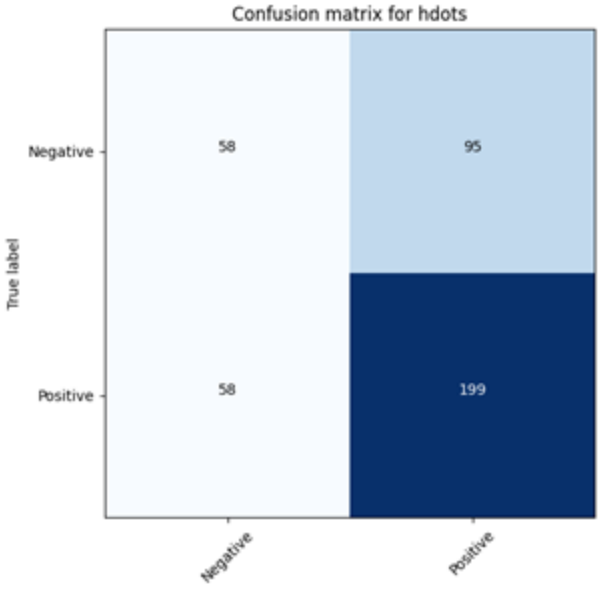


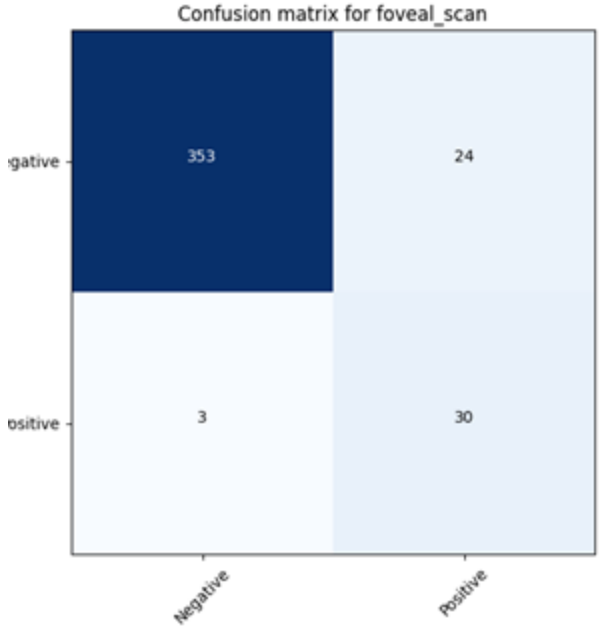


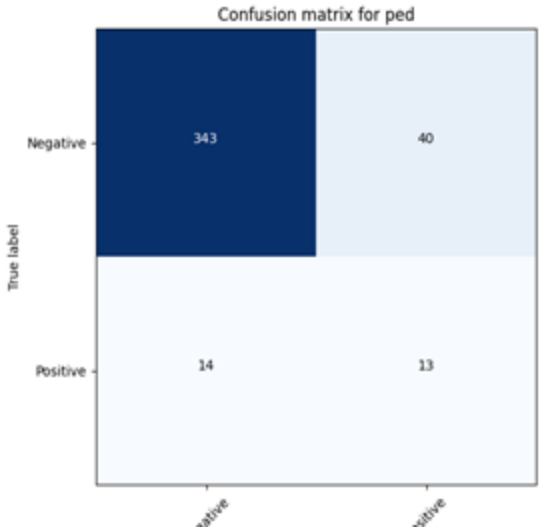


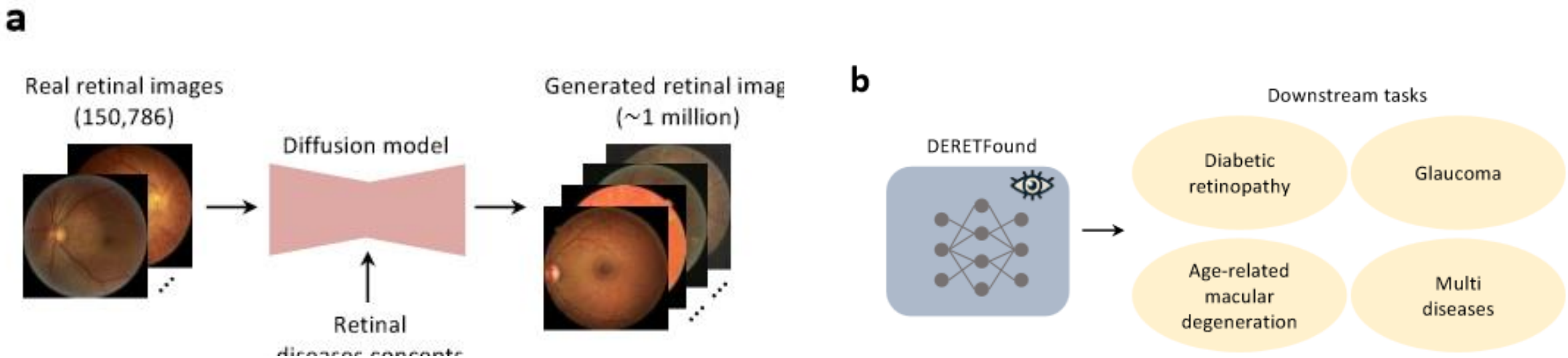
Separate RETFOUND - Confusion Matrices



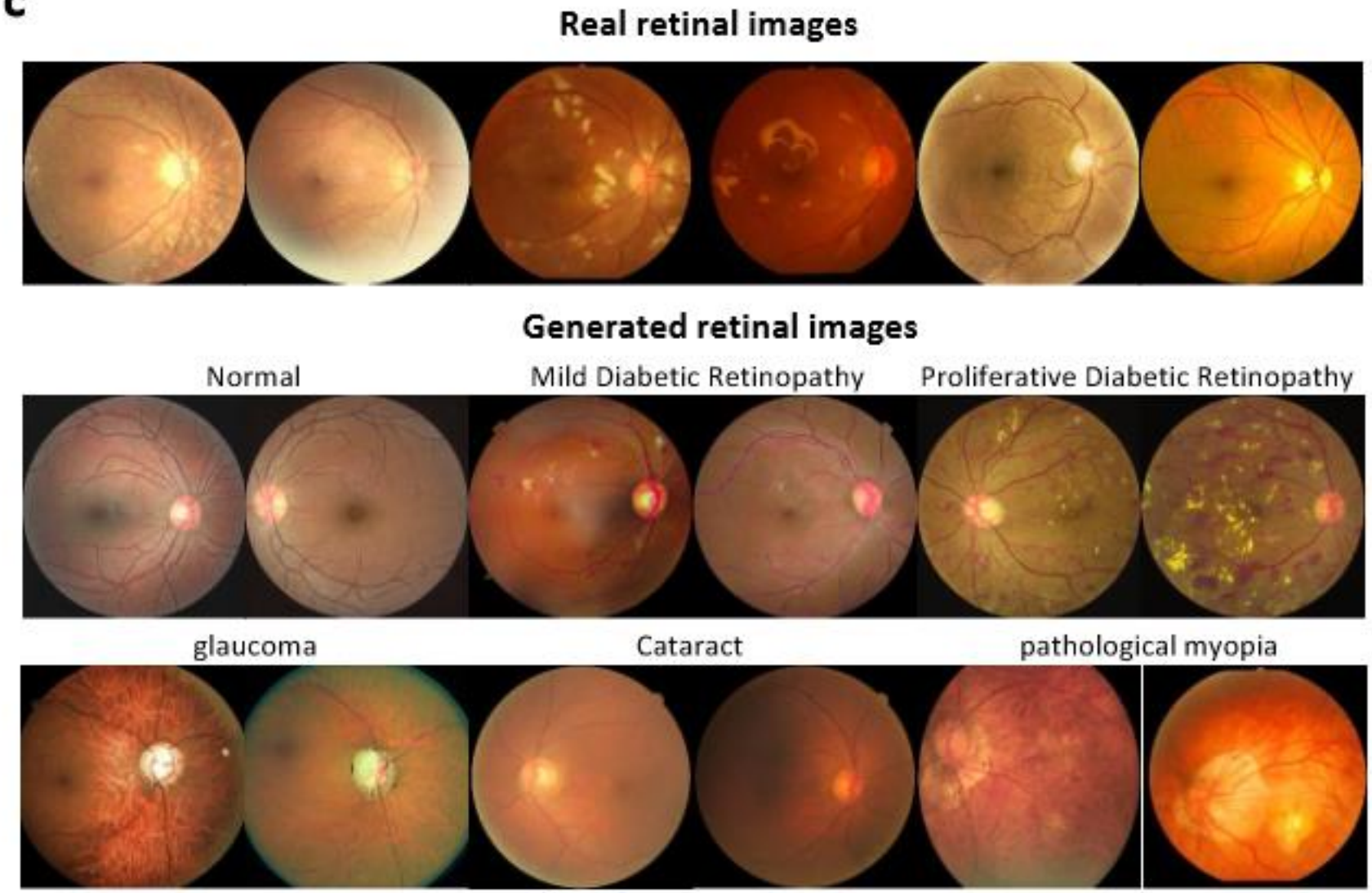


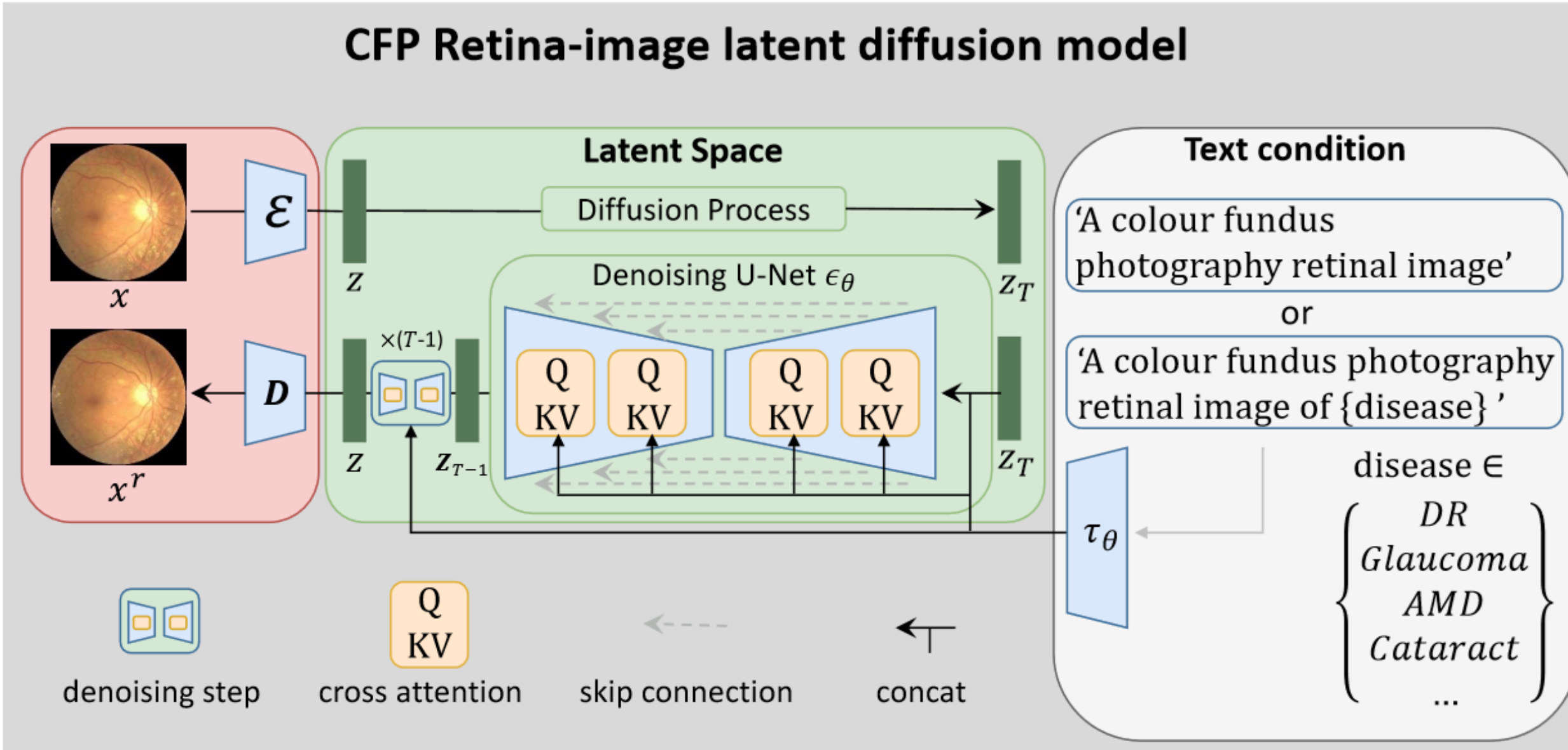


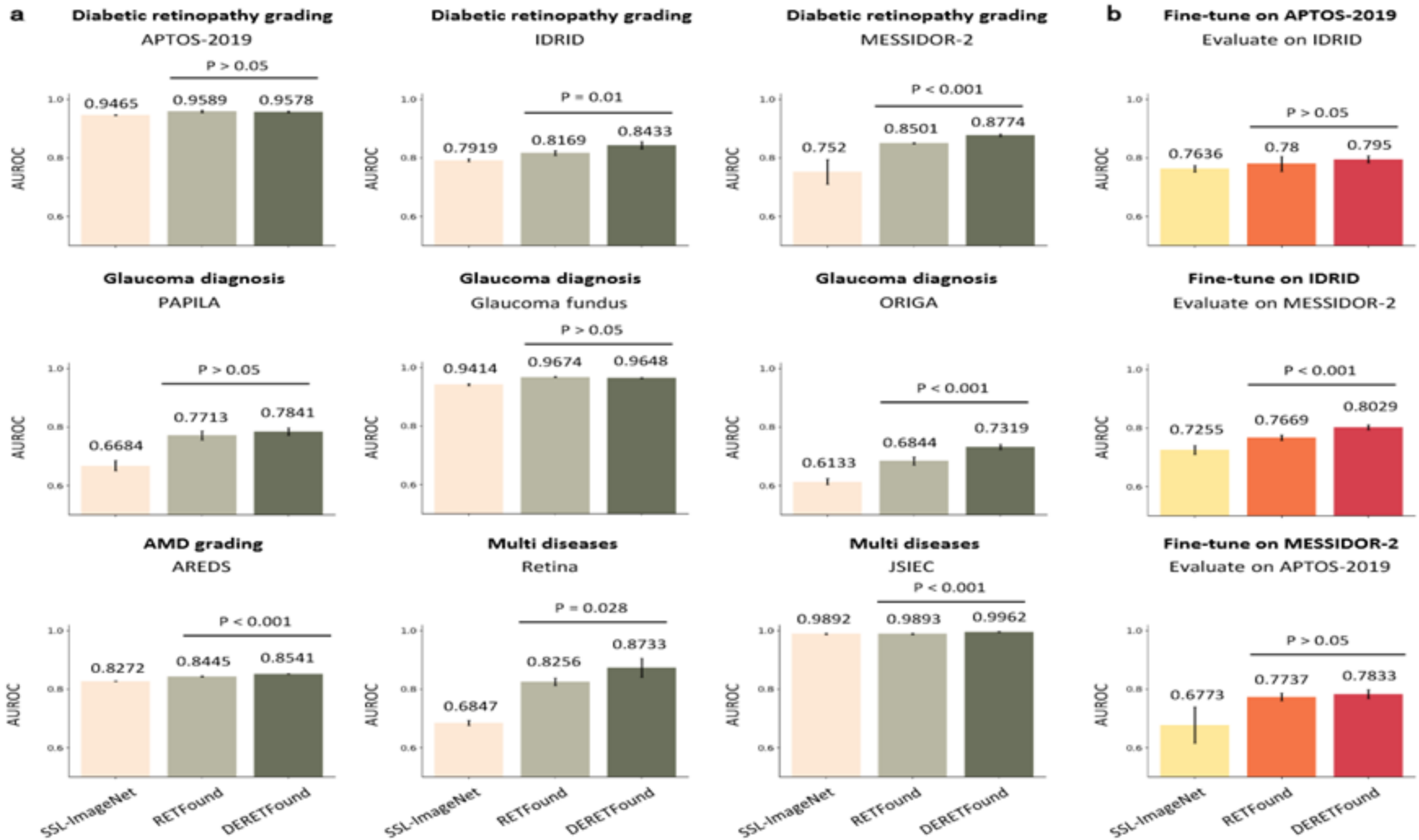


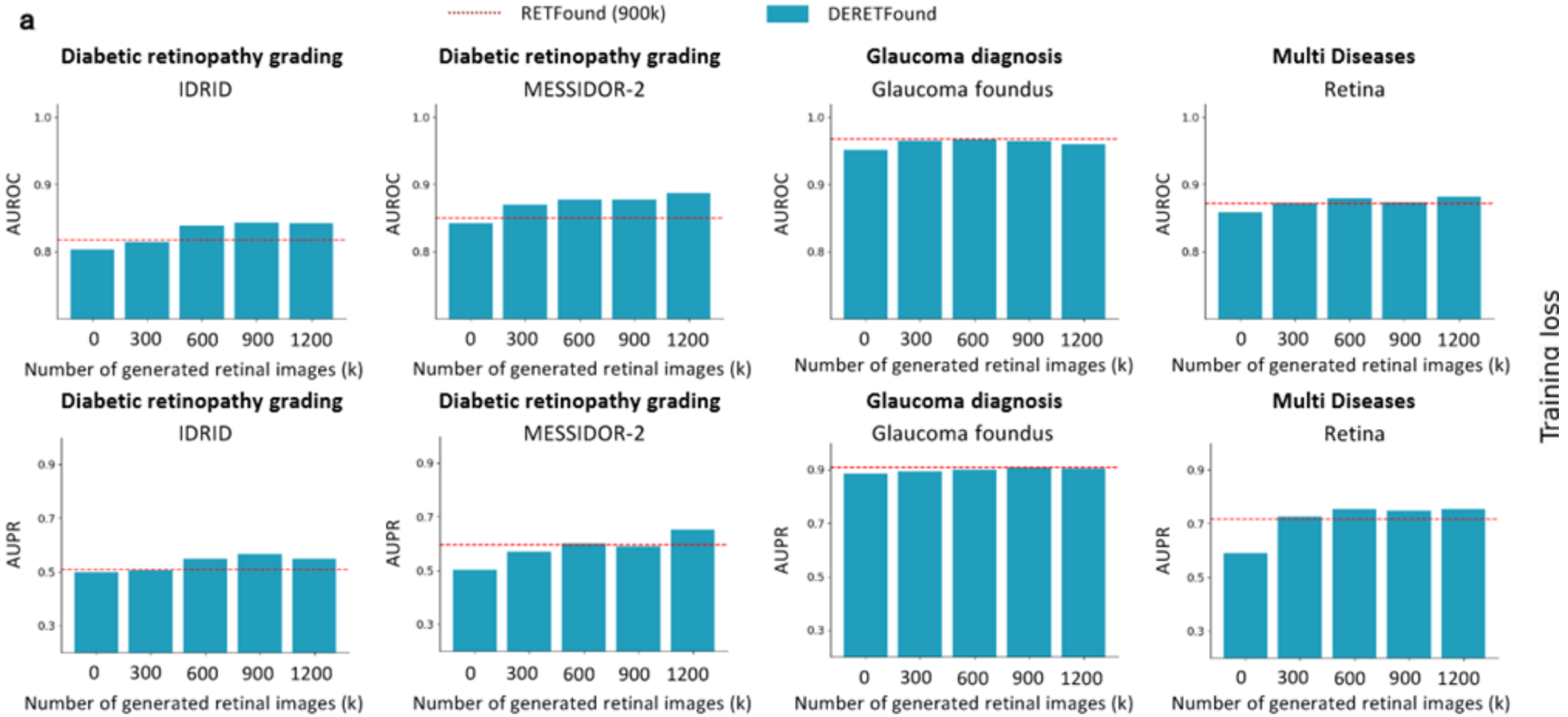

a
Real retinal images
(150,786)
Diffusion model
Retinal
Generated retinal imag
(~1 million)
b
DERETFound
Downstream tasks
Diabetic retinopathy
Glaucoma
Age-related macular degeneration
Multi diseases

c


Real retinal images
Generated retinal images
Normal
Mild Diabetic Retinopathy
Proliferative Diabetic Retinopathy
glaucoma
Cataract
pathological myopia

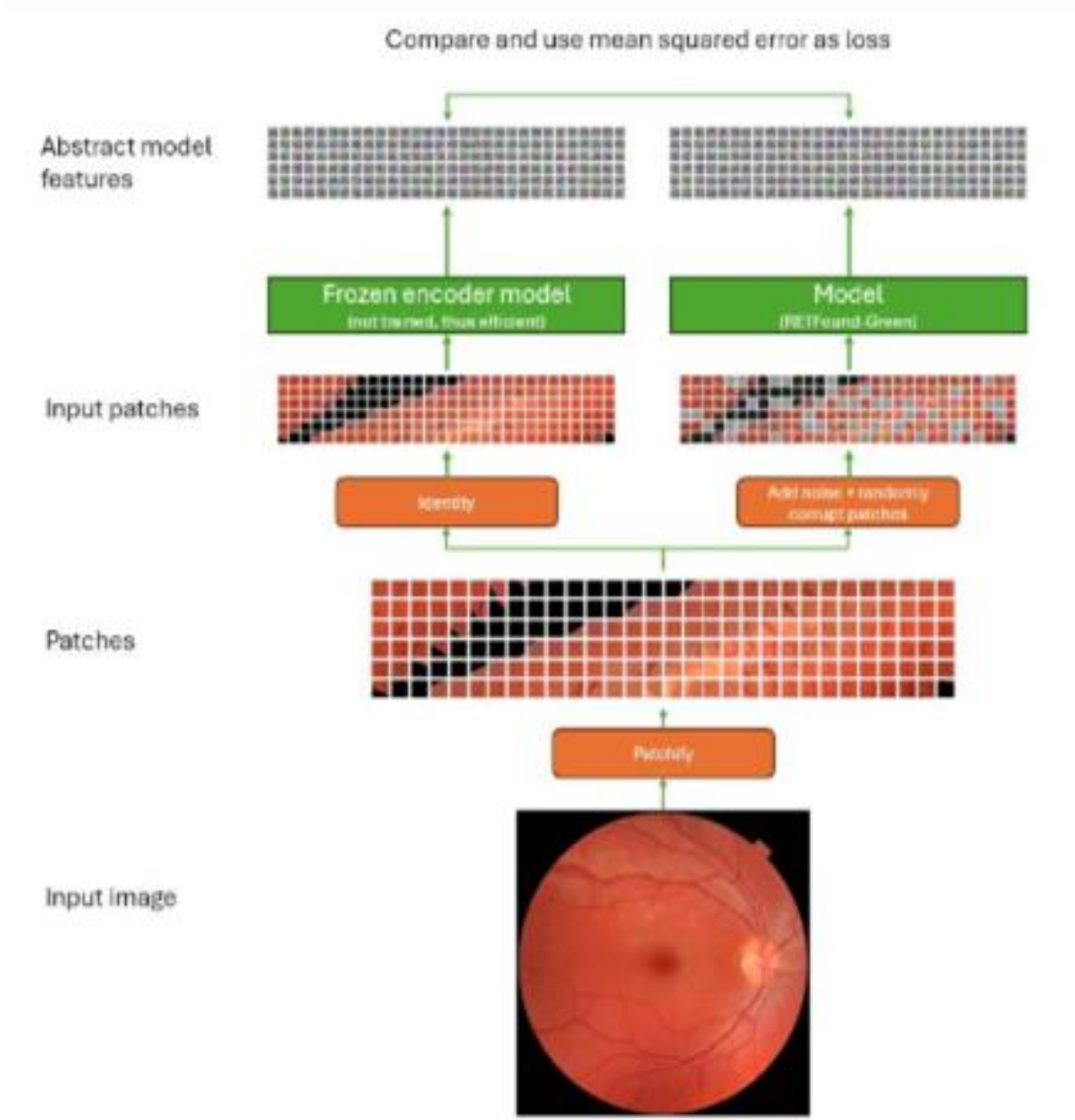

CFP Retina-image latent diffusion model
Latent Space
Diffusion Process
Denoising U-Net ϵ_θ
$\mathcal{E}$
D
x
x^r
z
z_T
z_{T-1}
×(T-1)
Q
KV
Text condition
'A colour fundus photography retinal image'
or
'A colour fundus photography retinal image of {disease}'
τ_θ
disease ∈
DR
$Glaucoma$
AMD
$Cataract$
...
denoising step
cross attention
skip connection
concat


a
Diabetic retinopathy grading
APTOS-2019
P > 0.05
0.9465
0.9589
0.9578
Diabetic retinopathy grading
IDRID
P = 0.01
0.7919
0.8169
0.8433
Diabetic retinopathy grading
MESSIDOR-2
P < 0.001
0.752
0.8501
0.8774
b
Fine-tune on APTOS-2019
Evaluate on IDRID
P > 0.05
0.7636
0.78
0.795
Glaucoma diagnosis
PAPILA
P > 0.05
0.6684
0.7713
0.7841
Glaucoma diagnosis
Glaucoma fundus
P > 0.05
0.9414
0.9674
0.9648
Glaucoma diagnosis
ORIGA
P < 0.001
0.6133
0.6844
0.7319
Fine-tune on IDRID
Evaluate on MESSIDOR-2
P < 0.001
0.7255
0.7669
0.8029
AMD grading
AREDS
P < 0.001
0.8272
0.8445
0.8541
Multi diseases
Retina
P = 0.028
0.6847
0.8256
0.8733
Multi diseases
JSIEC
P < 0.001
0.9892
0.9893
0.9962
Fine-tune on MESSIDOR-2
Evaluate on APTOS-2019
P > 0.05
0.6773
0.7737
0.7833
AUROC
1.0
0.8
0.6
SSL-ImageNet
RETFound
DERETFound

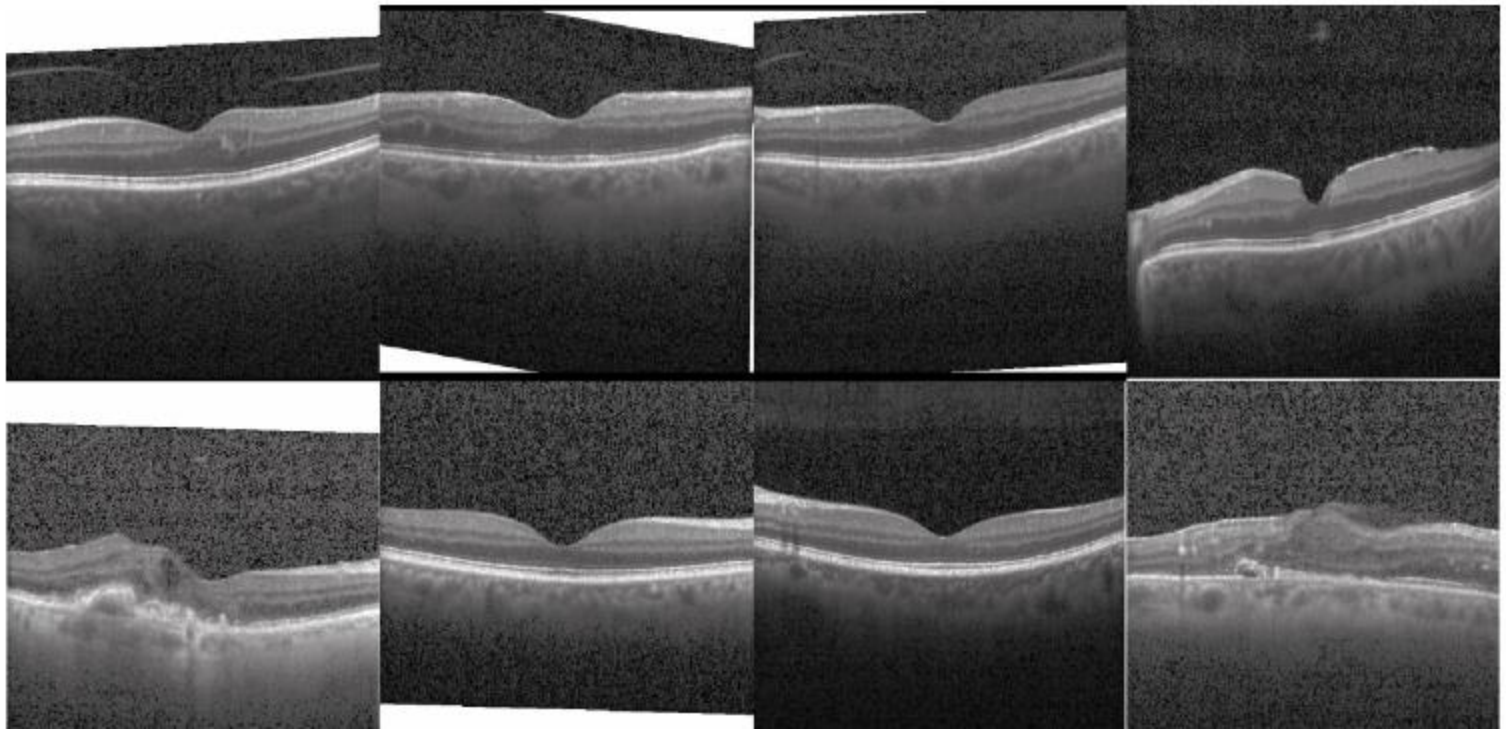

a
RETFound (900k)
DERETFound
Diabetic retinopathy grading
IDRID
AUROC
Diabetic retinopathy grading
MESSIDOR-2
AUROC
Glaucoma diagnosis
Glaucoma foundus
AUROC
Multi Diseases
Retina
AUROC
Diabetic retinopathy grading
IDRID
AUPR
Diabetic retinopathy grading
MESSIDOR-2
AUPR
Glaucoma diagnosis
Glaucoma foundus
AUPR
Multi Diseases
Retina
AUPR
0 300 600 900 1200
Number of generated retinal images (k)
Training loss

RETFOUND-Green

	RETFound-MEH	DERETFound	RETFound-Green	Green vs best
Training data	904,170 images	150,786 images	75,000 images	2x less
Training compute	112 A100 days	163 A100 days	~0.27 A100 days	>400x less
Training cost (monetary/carbon, estimate)	~$10,000 / 81kg of coal burned	~$14,000 / 117kg of coal burned	<$100 / 0.2kg of coal burned	>100x less
Training hardware	8x top datacentre GPUs (total VRAM: 320 GB)	8x top datacentre GPUs (total VRAM: 640 GB)	1x top consumer gaming GPU (total VRAM: 24 GB)	>8x less
Disk space (model)	1.12 GB (our optimisation, 3.68 GB originally)	1.12 GB (our optimisation, 3.68 GB originally)	0.09 GB	14x less
Disk space (1 million embeddings)	39.1 GB	39.1 GB	14.6 GB	2.6x less
Inference speed (same hardware)	6 img / s	6 img / s	16 img / s	2.7x faster
Linear probe speed (same hardware)	2.45 s / task	2.40 s / task	0.96 s / task	2.5x faster
Performance (Only counting wins with p<0.05)	At least comparable (Various BRSET tasks [Fig. 2]: 5 wins for RETFound-Green, 4 wins for DERETFound, none for RETFound-MEH. Diabetic retinopathy grading [Fig. 3]: 9 wins for RETFound-Green, 2 wins each for DERETFound and RETFound-MEH.)			Not generally inferior

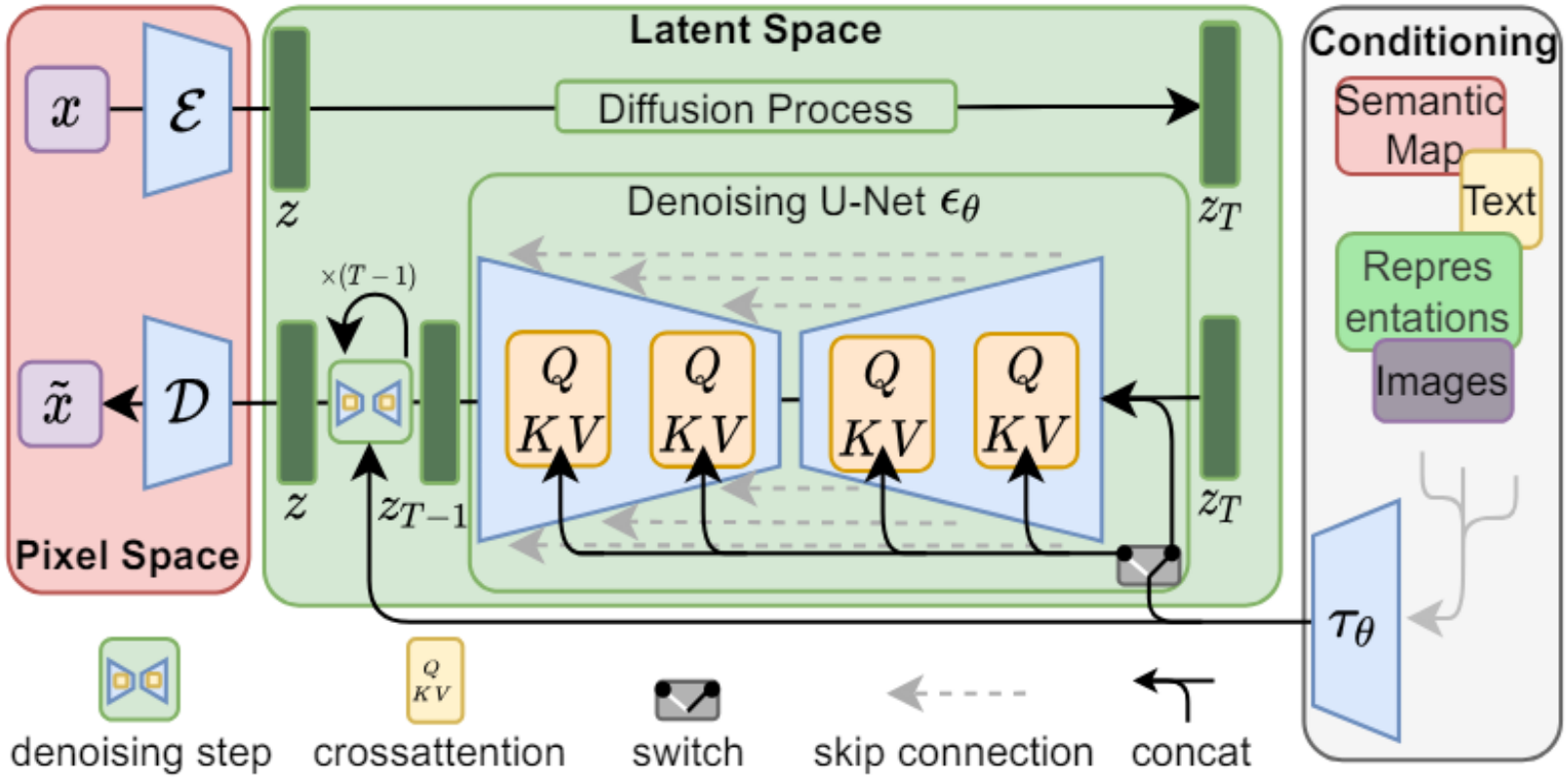

Compare and use mean squared error as loss
Abstract model features
Frozen encoder model
(not trained, thus efficient)
Model
(RETFound-Green)
Input patches
Identity
Add noise + randomly corrupt patches
Patches
Patchify
Input image

Diffusion Models



generated using MONAI Framework - Latent Diffusion Models

Latent Diffusion Models



Forward Process

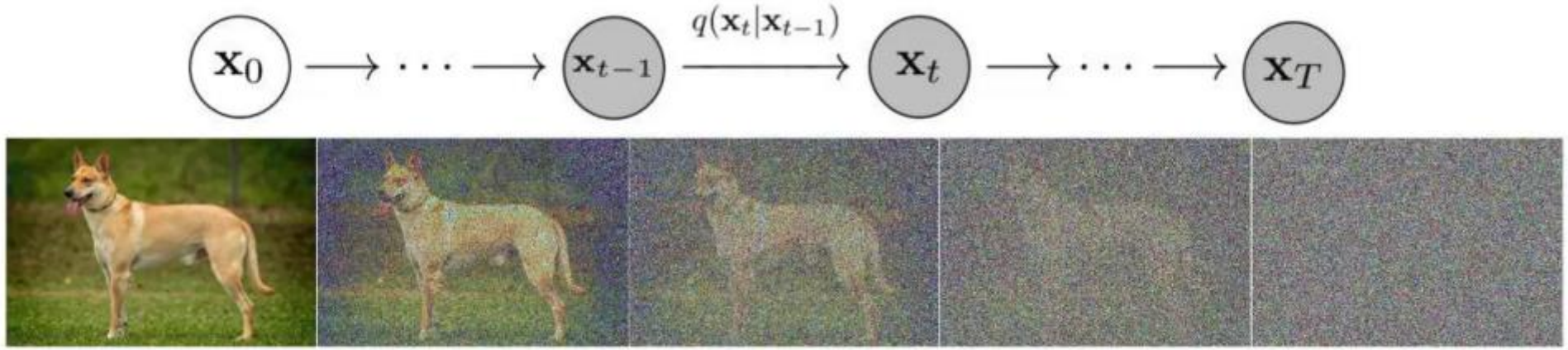


Fig 1. Forward Process

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I})$$

Reverse Process

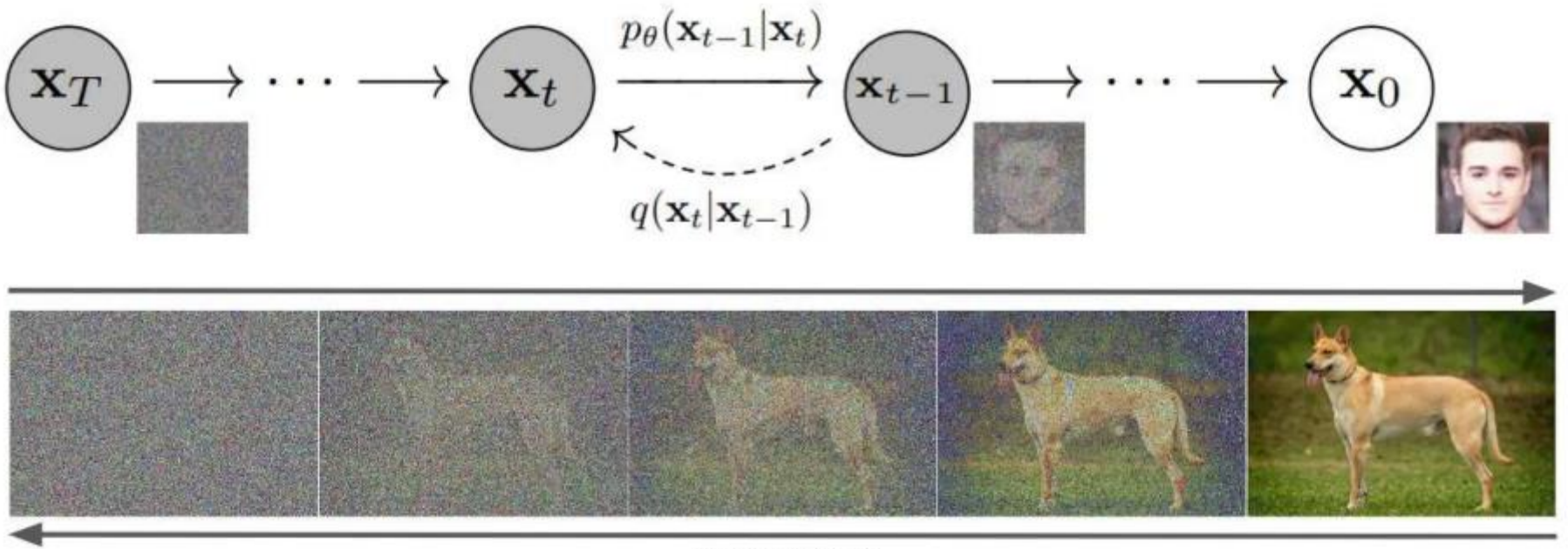
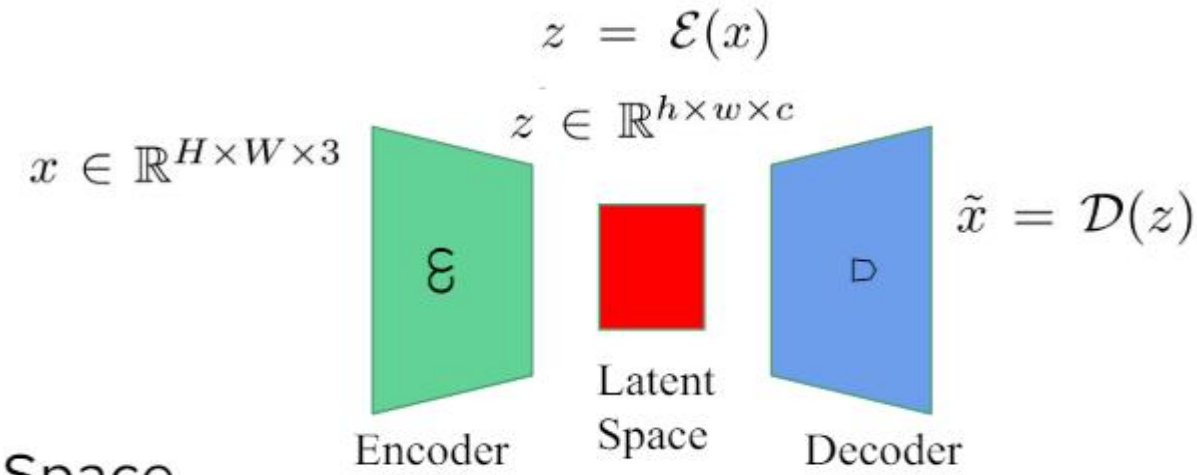


Fig 2. Reverse Process

$$p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \boldsymbol{\Sigma}_\theta(\mathbf{x}_t, t))$$

Method

- Pixel space
  - Computational overhead
- Solution?
  - Pixel Space -> Learned Latent Space



$$f = \frac{H}{h} = \frac{W}{w}$$

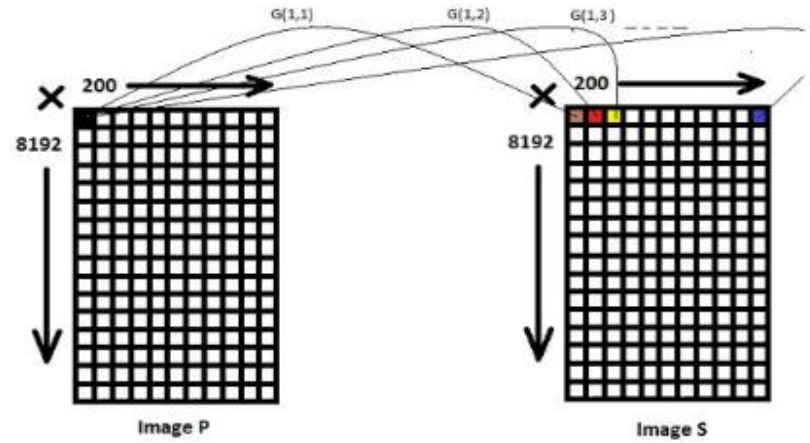
$$L_{\text{Autoencoder}} = \min_{\mathcal{E},\mathcal{D}} \max_{\psi} \Big(L_{rec}(x, \mathcal{D}(\mathcal{E}(x))) - L_{adv}(\mathcal{D}(\mathcal{E}(x))) + \log D_{\psi}(x) + L_{reg}(x; \mathcal{E}, \mathcal{D}) \Big)$$

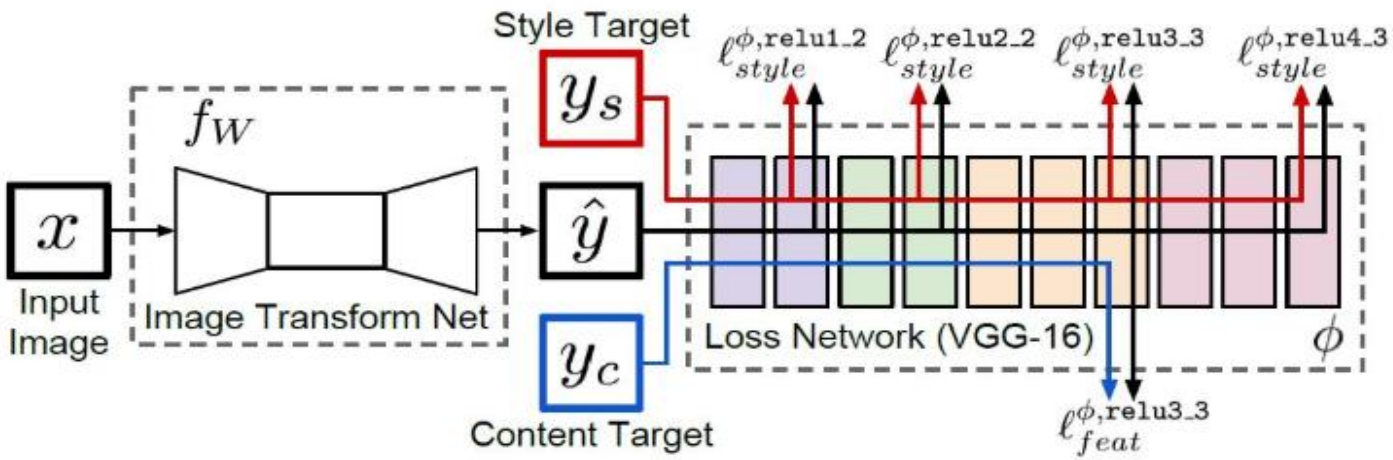
Reconstruction Loss Adversarial Loss Regularization

https://www.crcv.ucf.edu/wp-content/uploads/2018/11/Paper-3_-High-Resolution-Image-Synthesis-with-Latent-Diffusion-Models.pdf

Method - Reconstruction Loss

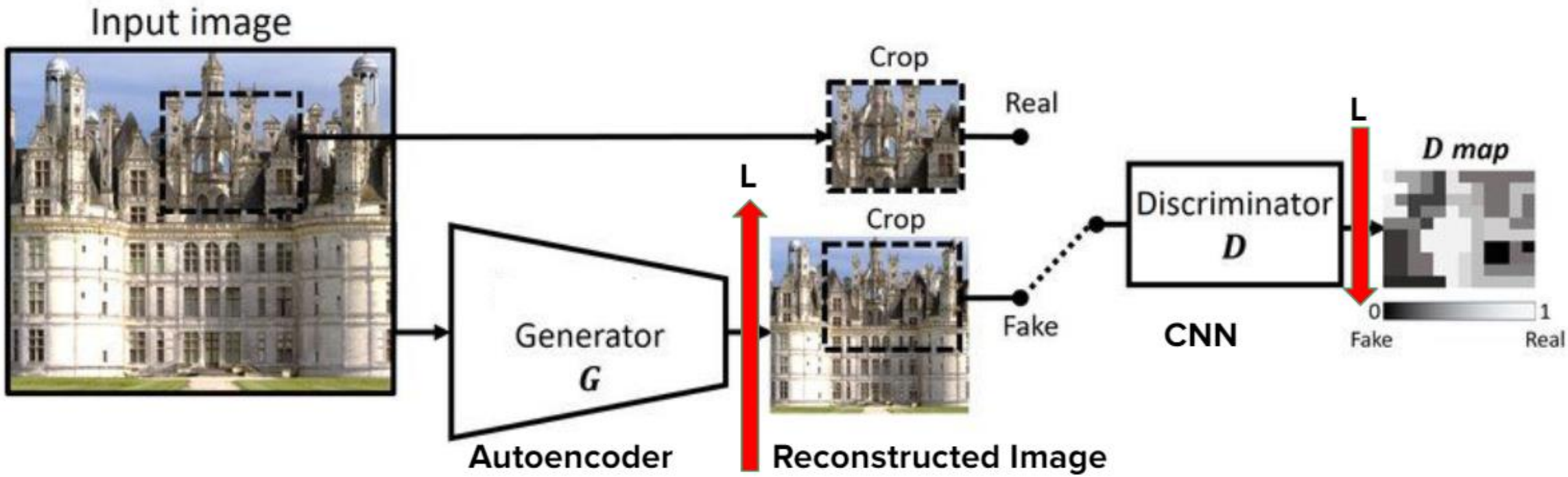
- L1/L2 + Perceptual
- Why Perceptual Loss
  - Style
  - Features
- How Perceptual Loss Works?





$$\ell^{\phi,j}_{feat}(\hat{y}, y) = \frac{1}{C_j H_j W_j} \|\phi_j(\hat{y}) - \phi_j(y)\|_2^2$$

Method - Adversarial Loss



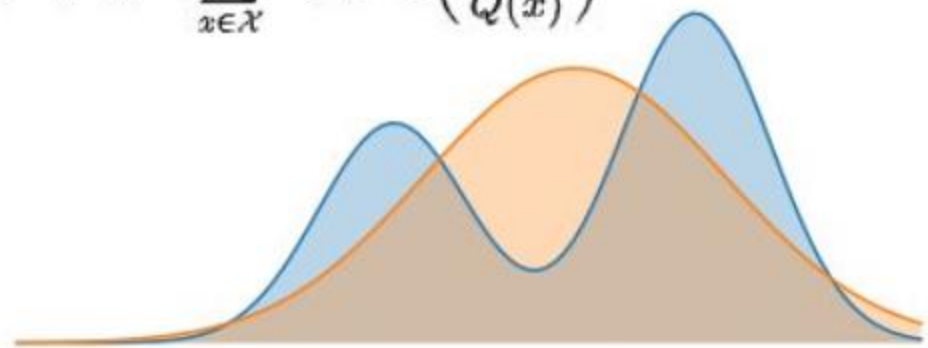
$$-L_{adv}(\mathcal{D}(\mathcal{E}(x))) + \log D_\psi(x)$$

$$-L_{adv}(\mathcal{D}(\mathcal{E}(x))) - (-\log D_\psi(x))$$

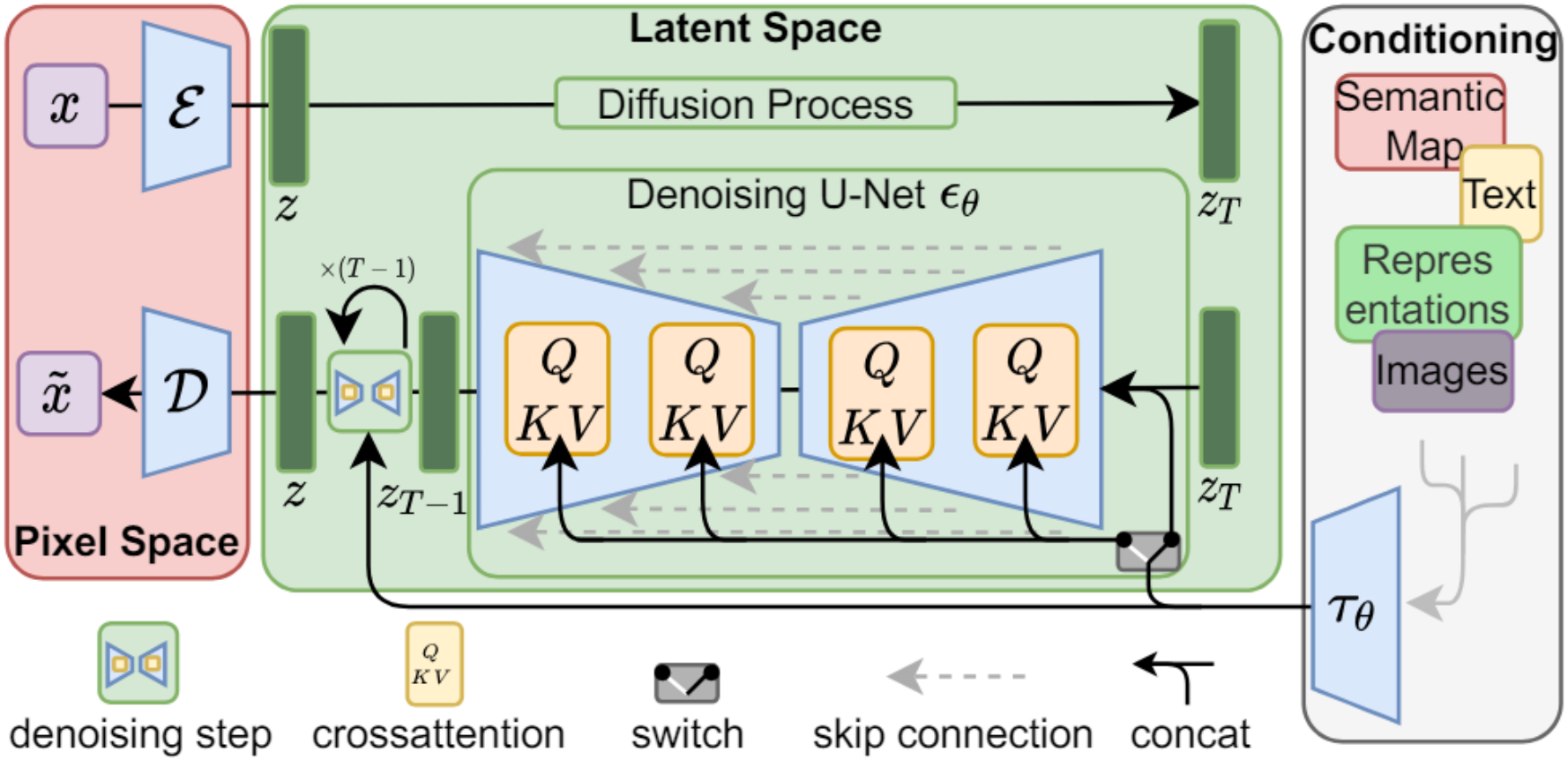
Method - Regularization

- KL-Reg (Kullback Leibler)
  - Distribution Similarity

$$D_{\mathrm{KL}}(P \parallel Q) = \sum_{x \in \mathcal{X}} P(x) \log\left(\frac{P(x)}{Q(x)}\right).$$



- VQ-Reg (Vector Quantized)
  - Discrete Latent Representation



$$L_{DM} = \mathbb{E}_{x,\epsilon\sim\mathcal{N}(0,1),t}\left[\|\epsilon - \epsilon_\theta(x_t, t)\|_2^2\right]$$

$$L_{LDM} := \mathbb{E}_{\mathcal{E}(x),\epsilon\sim\mathcal{N}(0,1),t}\left[\|\epsilon - \epsilon_\theta(z_t, t)\|_2^2\right]$$

$$\tau_\theta(y) \in \mathbb{R}^{M \times d_\tau}$$

$$Q = W_Q^{(i)} \cdot \varphi_i(z_t)$$

$$K = W_K^{(i)} \cdot \tau_\theta(y)$$

$$\varphi_i(z_t) \in \mathbb{R}^{N \times d_\epsilon^i}$$

$$W_K^{(i)} \in \mathbb{R}^{d \times d_\tau}$$

$$W_Q^{(i)} \in \mathbb{R}^{d \times d_\tau}$$

$$V = W_V^{(i)} \cdot \tau_\theta(y)$$

$$W_V^{(i)} \in \mathbb{R}^{d \times d_\epsilon^i}$$

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK}{\sqrt{d}}\right) \cdot V,$$

$$L_{LDM} := \mathbb{E}_{\mathcal{E}(x), y, \epsilon \sim \mathcal{N}(0,1), t}\left[\|\epsilon - \epsilon_\theta(z_t, t, \tau_\theta(y))\|_2^2\right]$$

TASK

- **Predict Presence/Absence of features in the OCT Image**

- Healthy
- Foveal Scan
- Intraretinal Fluids
- Subretinal Fluids
- Drusen
- Pigment epithelium detachment
- Hyperreflective Dots
- Hyperreflective Foci

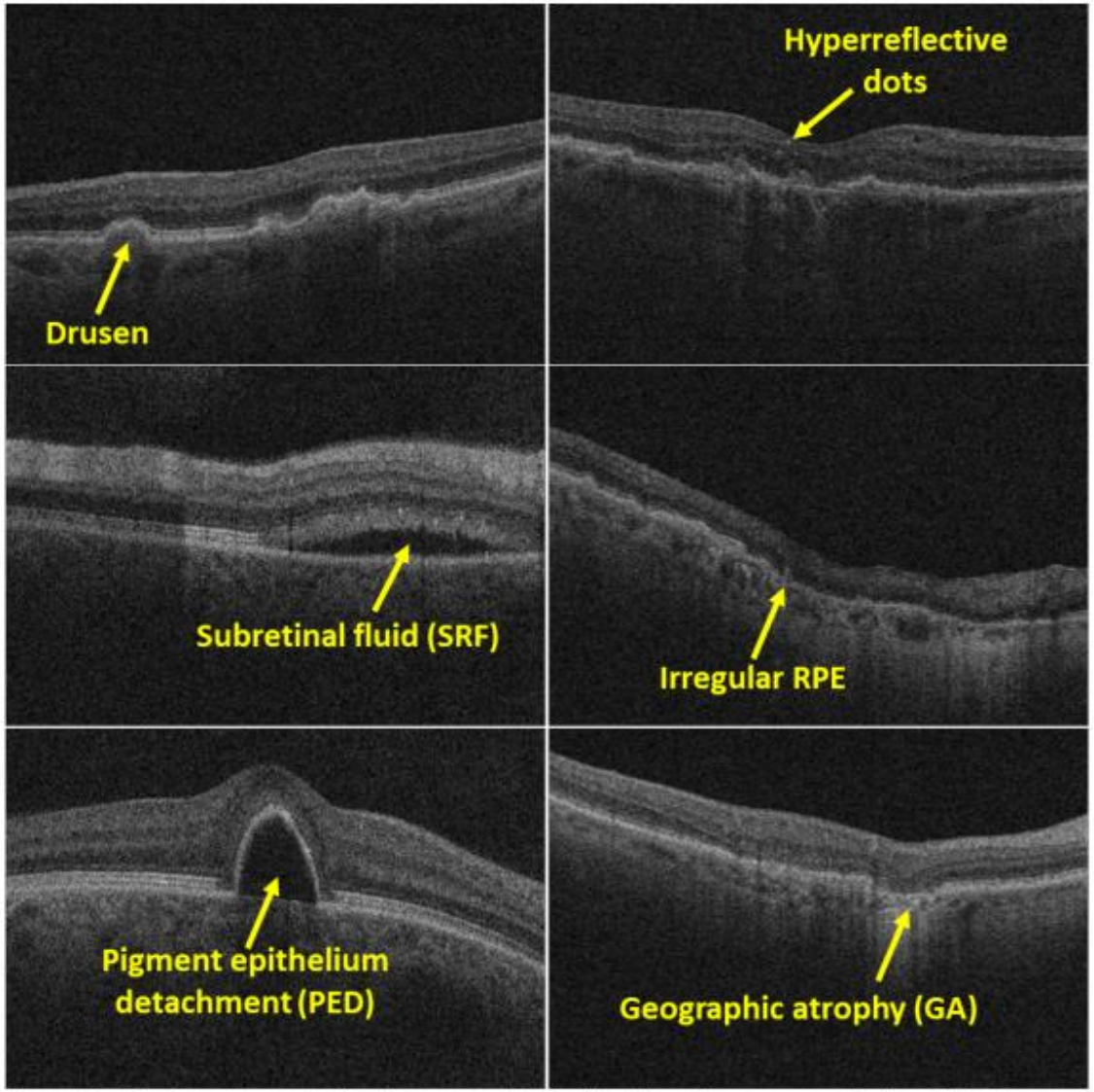


Figure 1: Various disease signatures manifested in the retina as seen in the OCT B-scans.

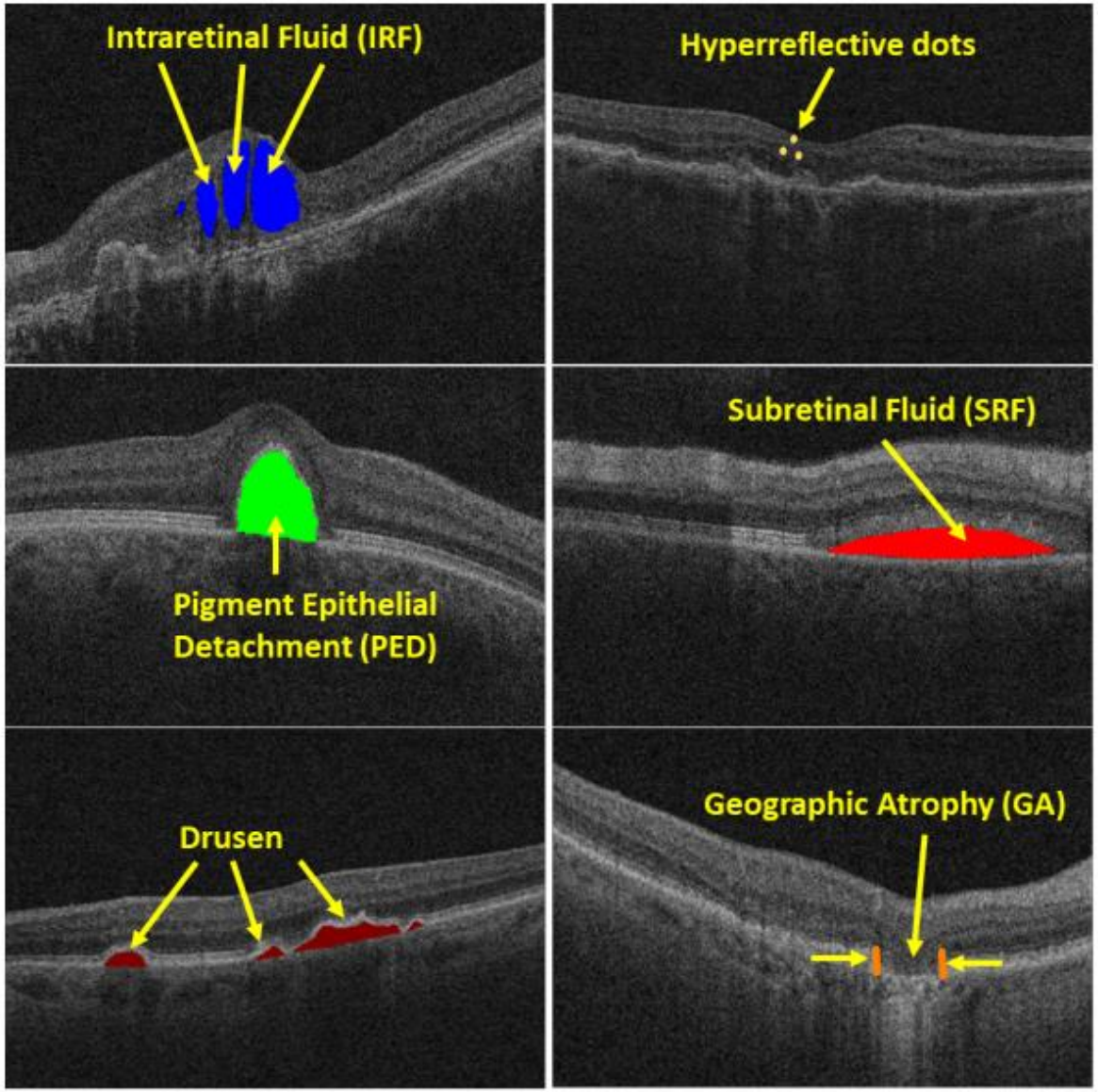
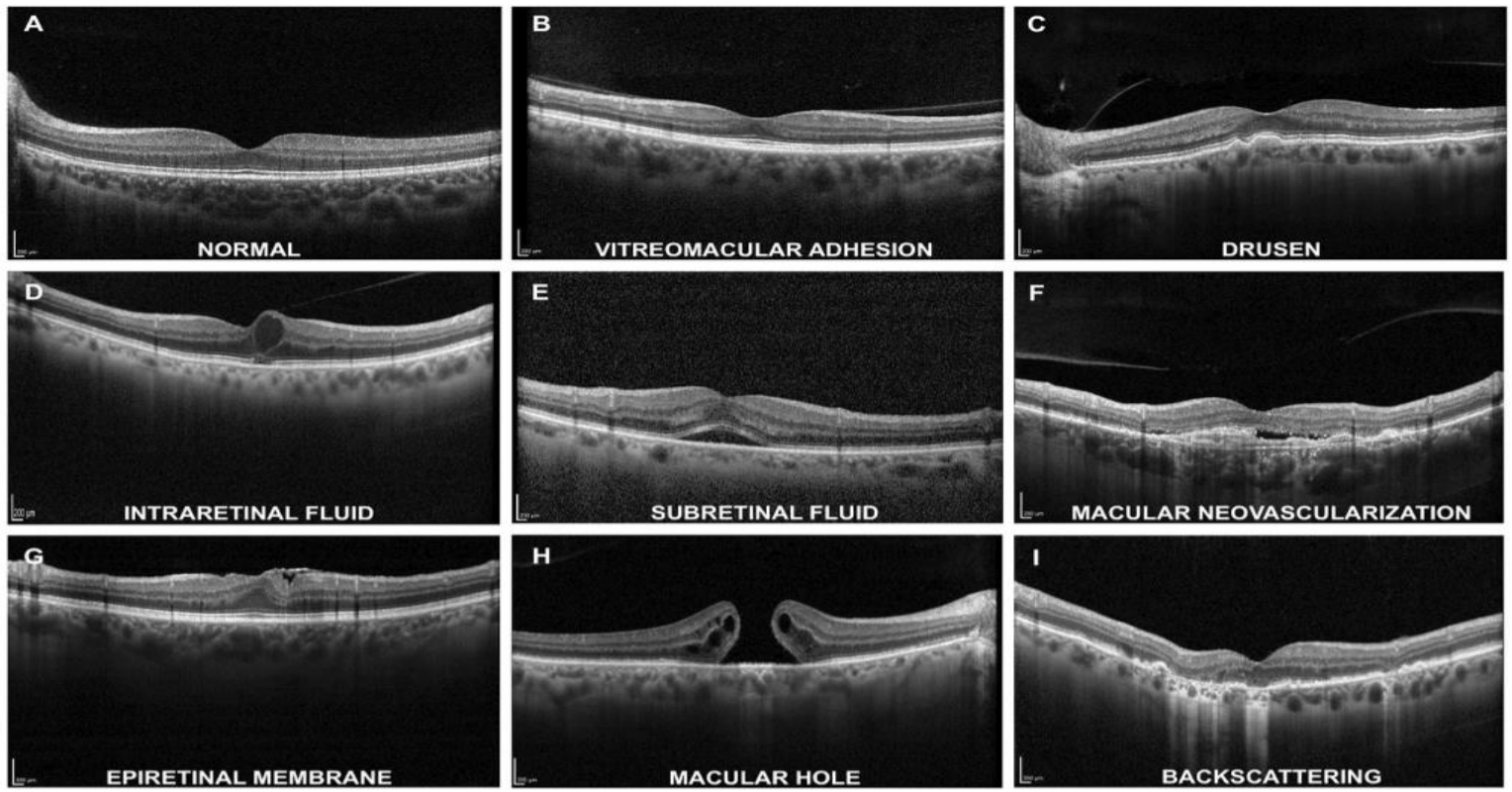


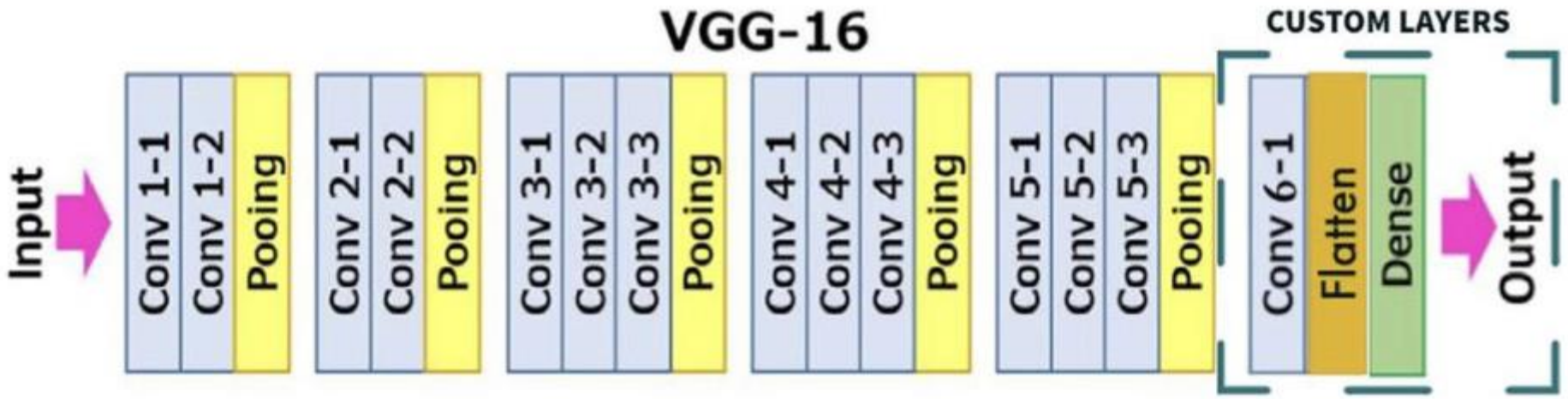
Figure 2: Manual delineations/markings for the disease signatures.

OCT-based deep-learning models for the identification of retinal key signs

Inferrera Leandro[1,4]✉, Borsatti Lorenzo[1,4], Miladinovic Aleksandar[2,4], Marangoni Dario[1], Giglio Rosa[1], Accardo Agostino[3] & Tognetto Daniele[1]



- Spectralis OCT (Heidelberg) with 815 nm laser source
- Axial resolution - 3.9 µm/pixel
- Lateral resolution - 5.7 µm/pixel
- 768 × 496 pixel image
- Central Region cropped (621 x 445) and then resized to 224x224



- 1 model – to classify as healthy/pathological
- 8 separate models to predict the absence of retinal signatures

Results

	Healthy	Pathological
Healthy	309	3
Pathological	6	306

	ERM	O.S.
ERM	249	4
O.S.	5	248

	IF	O.S.
IF	193	5
O.S.	0	197

	SF	O.S.
SF	68	2
O.S.	1	69

	D	O.S.
D	154	8
O.S.	10	152

	MNV	O.S.
MNV	58	6
O.S.	3	61

	VMA	O.S.
VMA	199	0
O.S.	3	197

	MH	O.S.
MH	46	2
O.S.	0	48

	BS	O.S.
BS	79	2
O.S.	0	83

Table 3. Confusion matrices obtained on the test set for each model: Healthy vs Pathological, One sign (ERM, IF, SF, D, MNV, VMA, MH or BS) vs all Other Signs (O.S.).

	Accuracy	Sensitivity	Specificity	Kappa	AUC
Healthy	0.99	1.00	0.99	0.98	0.99
ERM	0.97	0.96	0.98	0.94	0.97
IF	0.99	0.98	0.99	0.97	0.99
SF	0.96	0.94	0.99	0.93	0.96
D	0.96	0.96	0.96	0.92	0.96
MNV	0.95	0.92	0.97	0.90	0.95
VMA	0.99	1.00	0.98	0.98	0.99
MH	0.98	0.96	0.99	0.95	0.98
BS	0.93	0.91	0.95	0.86	0.93