Supplementary data

Supplementary Appendix 1. Implementation detail about coronary artery recognition network

Model structure

Existing DNN algorithms usually predict the category of each pixel. The pixel-level accuracy may be high, but the relationship between pixels is easily overlooked, making the vessel segmentation results discontinuous. Thus, we modified a special DNN: conditional generative adversarial network (cGAN) **(Supplementary Figure 4)** for image segmentation (so-called pix2pix, pix2pixHD). This cGAN consists of a generator and a discriminator. The training process of this cGAN can be treated as a competitive procedure between the generator and the discriminator. In the end, the entire model reaches Nash equilibrium. In the evaluation process, we only apply the generator to generate artery recognition results. The generator takes the coronary angiogram input and outputs the coronary artery category to which each pixel belongs. We rearrange the prediction results of each pixel into an image, indicating the recognition result of DNN. The size of the input angiogram and the output image result is 512×512 pixels.

In the generator, we apply the U-net structure with four down-sampling blocks (down-sample the input from 512x512 to 32x32) and four up-sampling blocks (up-sample the input from 32x32 to 512x512) as the generator part, which is shown in **Supplementary Figure 4**. The discriminator part contains three sub-discriminators to discriminate on three different scales and average the results. The three distinguishing scales are the original image, 1/2 down-sampling of the original image, and 1/4 down-sampling of the original image. These three layers build an image pyramid and train a discriminator for each layer. It is notable that we use a convolutional "PatchGAN" classifier in these sub-discriminators, which only penalises structure at the scale of image patches. More specifically, each sub-discriminator outputs an 8×8 matrix. Each element of this matrix is a single value (from 0 to 1) corresponding to a 64×64 patch of input. Each sub-discriminator tries to classify if each 64×64 patch in an image is real or fake. We run this discriminator convolutionally across the image, averaging all responses to provide the ultimate output of sub-discriminator. This design can significantly improve the spatial continuity of segmentation results.

In the network testing process, the outputs of the generator are treated as segmentation results. The output has three RGB channels (a GAN model output has the same shape as its input). Each pixel in the output will be converted to a prediction label according to the Euclidean distance between pixel value and prediction label value. The mapping relationship between prediction label value and coronary artery segments is shown in **Supplementary Table 4**. For example, a predicted pixel (250,249,248) will be converted to the label value (255,255,255), which has the minimal Euclidean distance to this predicted pixel. Different kinds of ground-truth triad value only contain 0,128 or 255, which ensures the large Euclidean distance between different ground-truth values so that each pixel will eventually converge to its ground-truth label. However, the model will publish various kinds of misclassified pixels differently in the initial stage of network training, which slows down the network training.

Multiple-channel output (21 one-hot channels) may improve the issue even though it increases the amount calculation slightly. In this work, we set the output as 3 RGB image.

Loss function

(1) GAN loss.

The optimisation process of our Conditional GANs can be described as the following minimax game:

$$\min_G \max_D L_{GAN}(G, D),$$

Where the loss function $L_{GAN}(G, D)$ is given by

$$E_{(i,o)}[logD(i, o)] + E_{(i)}[\log(1 - D(i, G(i)))],$$

where i is an angiogram and o is a segmentation result image. Our discriminator part has three sub-discriminators. The learning problem then becomes a multi-task learning problem of

$$\min_G \max_{D_1 D_2 D_3} \sum_{k=1,2,3} L_{GAN}(G, D_k).$$

(2) Feature matching loss.

In each sub-discriminator, we calculated the pixel-wise loss between the feature of the generated segmentation result by generator and the feature of the ground truth. The loss is shown as:

$$L_{FM}(G, D_k) = \sum_{i=1}^{T} \frac{1}{N_i} \left\| D_k^{(i)}(i, o) - D_k^{(i)}(i, G(i)) \right\|_1,$$

where *T* is the total number of layers (herein there are five layers in each sub-discriminator), N_i denotes the number of elements in each layer, $D_k^{(i)}$ denotes the ith-layer feature extractor of sub-discriminator D_k.

(3) VGG-based perceptual loss.

Like feature matching loss, we also extract the VGG feature of the generated segmentation result by generator and the VGG feature of the ground truth, considering the low-level feature (edge and context) extracted by pre-trained VGG network, also avails angiogram analysis. The loss is shown as:

$$L_{VGG}(G, D_k) = \sum_{j=1}^{N} \frac{1}{M_j} \left\| VGG^{(j)}(o) - VGG^{(j)}(G(i)) \right\|_1,$$

where N is the total number of VGG layers, M_j denotes the number of elements in each layer, $VGG_k^{(j)}$ denotes the jth-layer feature extractor of VGG.

The total loss function is the sum of the above three losses, shown as:

$$\min_G \max_{D_1 D_2 D_3} \sum_{k=1,2,3} L_{GAN}(G, D_k) + \alpha \sum_{k=1,2,3} L_{FM}(G, D_k) + \beta \sum_{k=1,2,3} L_{VGG}(G, D_k)$$

Implementation detail

For training, the initial learning rate is 2×10^5, which gradually dropped to 10^6 during training. For the weight parameter, we set α as 0.5 and β as 10. The number of training epochs is 400. Graphics processing units are NVIDIA GTX 1080Ti GPUs. Adam optimiser was used to optimise our model. Training time lasted about five days.

Supplementary Appendix 2. Implementation detail about lesion morphology detection network

Model structure

For the lesion morphology detection task, we developed a convolutional DNN **(Supplementary Figure 4)**, which takes coronary angiogram input and outputs the location (the upper left and lower right coordinates of the predicted rectangular area) and type (a scalar) of all the lesion morphologies that appeared in the input angiogram. Deep residual block, up-sampling layer, and lateral connection are used to extract different scale features of different lesions. Using these feature maps, the Region Proposal Network generates region proposal areas where lesion morphologies may occur. After that, features of every region proposal area are fed into convolutional layers and fully connected layers, to predict the type and location of lesion morphologies.

As shown in **Supplementary Figure 5**, the input angiograms are down-sampled by 2×2 pooling layers and up-sampled by 2×2 interpolated layers. Different down-sampling features and up-sampling features are combined by element-sum (light green arrows) operator. Convolutional layers (green arrows) extract four different scale detection features. Using these feature maps. RPN (region proposal network) generates region proposal areas where lesion morphologies may occur. After that, features of every region proposal area are fed into several convolutional layers and fully connected layers, to predict the type and location of lesion morphologies.

We treated four different scale detection features (dark blue blocks as F_0, F_1, F_2, F_3) as a feature pyramid and viewed it as if it were produced from an image pyramid. Thus, we can adapt the assignment strategy of region-based detectors (in Fast r-cnn) in the case when they run on image pyramids. Formally, we assign a region of interest (RoI) of width *w* and height *h* (on the input image to the network) to the level F_k of our feature pyramid by: $k = \lfloor k_0 + log_2(\sqrt{wh}/224) \rfloor$. Here k_0 is the target level on which an RoI with $w \times h = 224^2$ should be mapped into. We set k_0 to 3. Intuitively, the equation above means that if the RoI's scale becomes smaller (say, 1/2 of 224), it should be mapped into a finer-resolution level (say, k = 2).

Loss function

After assigning the detection feature for a RoI, the optimisation process of our lesion detection network can be described as:

$$L(p_i, t_i) = \frac{1}{N_{cls}} \sum_i L_{cls}(p_i, p_i^*) + \lambda \frac{1}{N_{reg}} \sum_i p_i^* L_{reg}(t_i, t_i^*)$$

Where p_i is the classification possibility of the ith anchor (the region proposal box generated by RPN, presented by a four tuple $\{x_1, y_1, x_2, y_2\}$, where (x_1, y_1) is the left-top corner of the box and (x_2, y_2) is the right-bottom corner of the box), we name the ith anchor as Anchor[i].

When the Anchor[i] is positive region proposal (the IOU of Anchor[i] and its Ground Truth Box >0.7), $p_i^* = 1$. When the Anchor[i] is negative region proposal (the IOU of Anchor[i] and its Ground Truth Box <0.3), $p_i^* = 0$. Anchors that are not positive or negative were not trained by the network. t_i is the

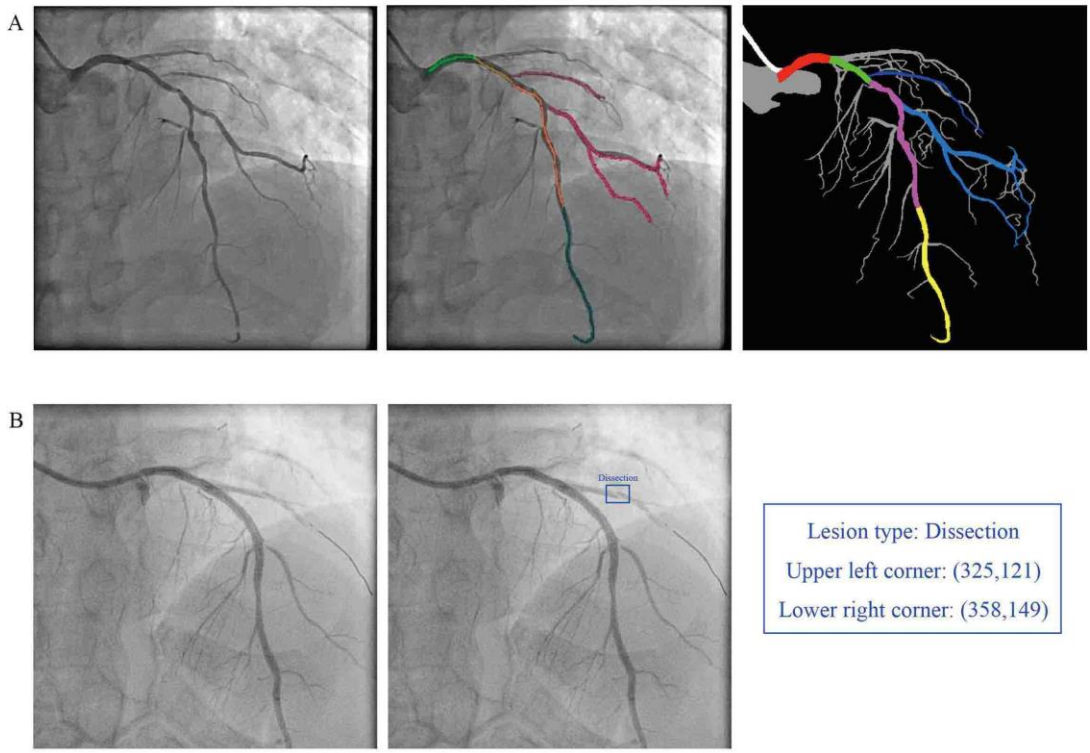
parameterised coordinates of the predicted box of Anchor[i], and t_i^* is the parameterised coordinates of the Ground Truth Box of Anchor[i].

For an anchor box, $t_i = \{t_x, t_y, t_w, t_h\}$ where $t_x = (x - x_a/w_a), t_y = (y - y_a/h_a), t_w = log(w/w_a), t_h = log(h/h_a)$; $t_i^* = \{t_x^*, t_y^*, t_w^*, t_h^*\}$ where $t_x^* = (x^* - x_a/w_a), t_y = (y^* - y_a/h_a), t_w = log(w^*/w_a), t_h = log(h^*/h_a)$, where (x, y) is the centre point of the predicted box, w, h are the weight and height of the predicted box, (x_a, y_a) is the centre point of the anchor box, w_a, h_a are the weight and height of the anchor box, (x^*, y^*) is the centre point of the Ground Truth box, w^*, h^* are the weight and height of the Ground Truth box.

N_{cls} is the minibatch size. N_{reg} is the number of anchor box. $L_{cls}(p_i, p_i^*) = -\log [p_i p_i^* + (1 - p_i)(1 - p_i^*)]$ and $L_{reg}(t_i, t_i^*) = Smooth_{L1}(t_i - t_i^*)$ where $Smooth_{L1}(x) = \begin{cases} 0.5 * x^2 & |x| < 1 \\ |x| - 0.5 & otherwise \end{cases}$.

Implementation details

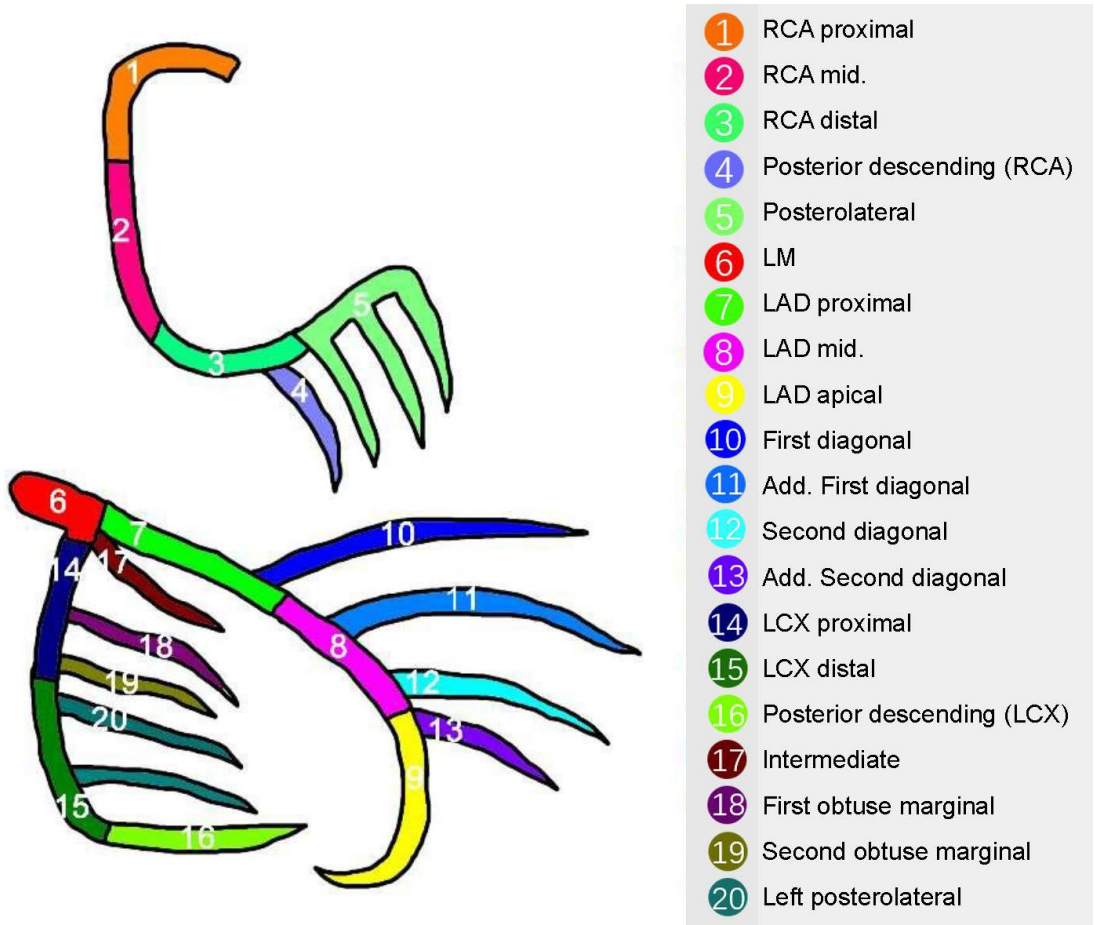
All architectures in **Supplementary Figure 4** are trained end to end. We adopt synchronised SGD optimiser to update our network. Graphics processing units are NVIDIA GTX 1080Ti GPUs. A mini-batch involves two images and 256 anchors per image. We use a weight decay of 0.0001, a momentum of 0.9 and a λ of 10. The learning rate is 0.02 for the first 30k mini-batches and 0.002 for the next 10k. The implementation details of FPN feature selection are set the same as in FPN (feature pyramid networks).



Supplementary Figure 1. The annotation procedure for coronary artery recognition and lesion morphology detection.

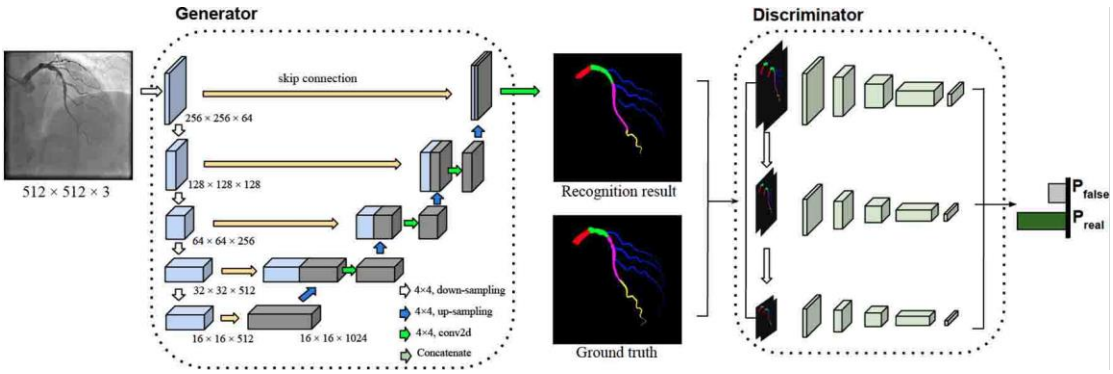
A) Coronary artery recognition. The first image is an original image. The second image is a sketchy-labelled image. The third image is a fine-labelled image.

B) Lesion morphology detection. The first image is an original image. The second image is a labelled image including lesion type and lesion location.



Supplementary Figure 2. Annotated coronary artery segments.

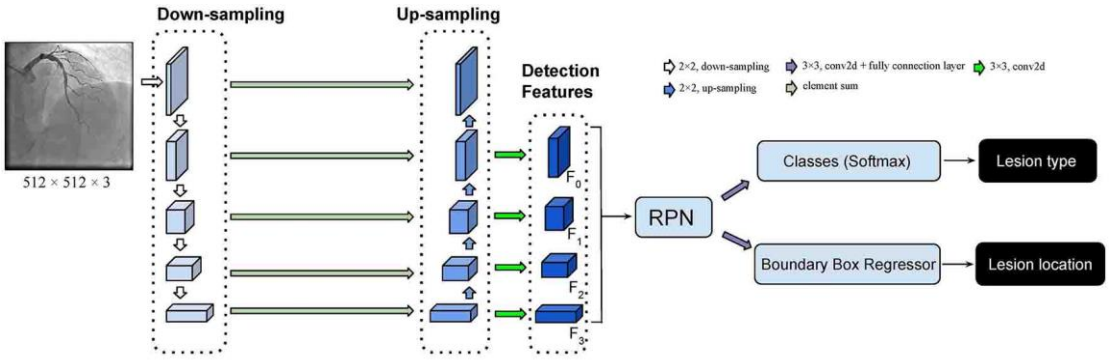
A total of 20 coronary artery segments were annotated in our study.



Supplementary Figure 3. The structure of the coronary artery recognition network.

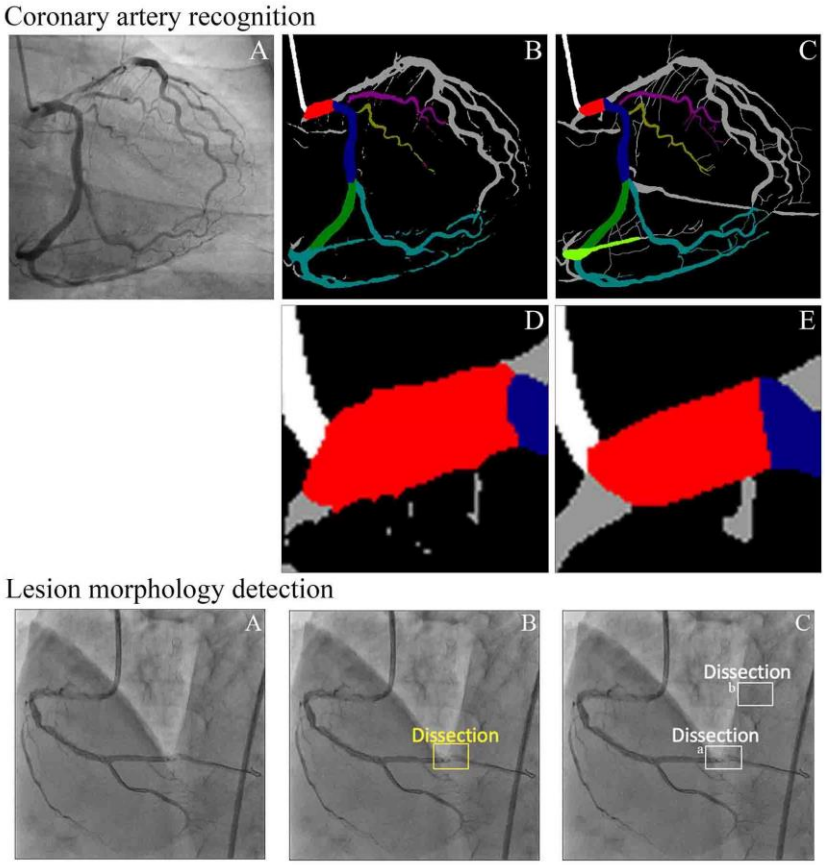
This GAN consists of a generator and a discriminator. The training process of this network can be treated as a competitive procedure between the generator and the discriminator. In the end, the entire model reaches Nash equilibrium and the accuracy of the discriminator is equal to 50%, which means that the discriminator is hard to discern the difference between recognition results and ground-truth images, showing that generator outputs a high-quality recognition result. In the evaluation process, we only apply the generator to generate artery recognition results.

In the generator, the input angiograms are down-sampled (white arrows) and then up-sampled (blue arrows) to generate features of different scale. These features are combined by concatenated operator (yellow arrows) to enrich the semantic information for better performance. In the discriminator, recognition results and ground truth are resized to different scale and processed and combined by several convolutional layers to generate the discrimination result.



Supplementary Figure 4. The structure of the lesion morphology detection network.

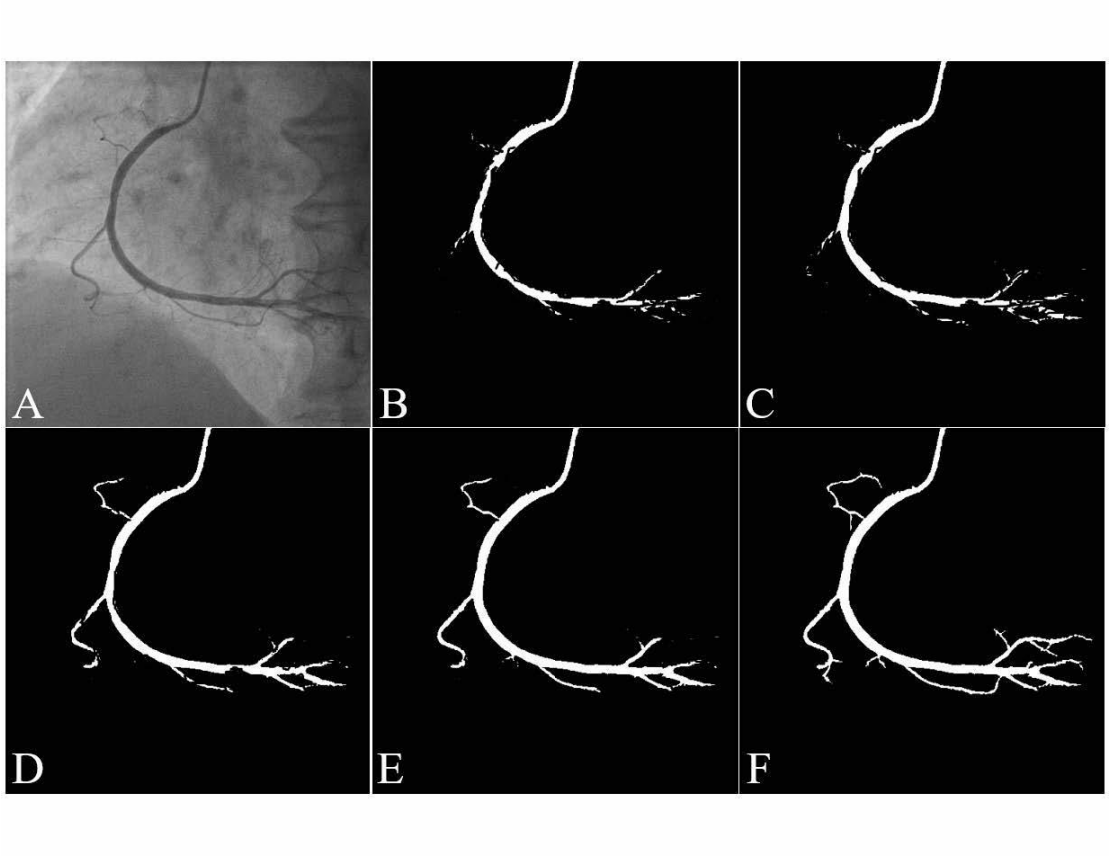
The input angiograms are down-sampled by 2x2 pooling layers and up-sampled by 2x2 interpolated layers. Different down-sampling features and up-sampling features are combined by element-sum (light green arrows) operator. Convolutional layers (green arrows) extract four different scale detection features. Using these feature maps, the RPN network generates region proposal areas where lesion morphologies may occur. After that, features of every region proposal area are fed into several convolutional layers and fully connected layers, to predict the type and location of lesion morphologies. RPN: region proposal network



Supplementary Figure 5. Evaluation process of the coronary artery recognition model and the lesion morphology detection model.

For the coronary artery recognition model, we feed an angiogram (A) as input image. The recognition model will output the results (C). The middle image is ground truth (B). D and E are generated by zooming in the LM segment in B and C. In the result image, the red area is the predicted area of the LM segment by the model. We counted the number of red pixels in this area (E) and ground-truth area (D) to calculate the pixel number of true positive (TP), true negative (TN), false positive (FP) and false negative (FN). The TP pixel is the red pixel in D and E. The TN pixel is the non-red pixel in D and E. The FP pixel is the red pixel in E but the non-red pixel in D. The FN pixel is the non-red pixel in E but the red pixel in D. Based on these results, we evaluate the segment recognition model by several metrics including accuracy ([TP+TN]/[TP+TN+FP+FN]), sensitivity (TP/[TP+FN]), specificity (TN/[TN+FP]), positive predictive value (TP/[TP+FP]), negative predictive value (TN/[TN+FN]). For the lesion morphology detection model, we feed an angiogram (A) as input. The detection model will output the results (C). The yellow bounding box and lesion name are ground truth (B). The white bounding boxes are detected lesion cases by models. One dissection lesion (a, overlap rate=0.90 >0.5) is detected correctly, and another one (b, overlap rate=0 <0.5) is detected incorrectly. For this angiogram, the precision rate is 50%, the recall rate is 100% and the F1 score is 0.667. We counted all detected lesion cases by models and all correctly

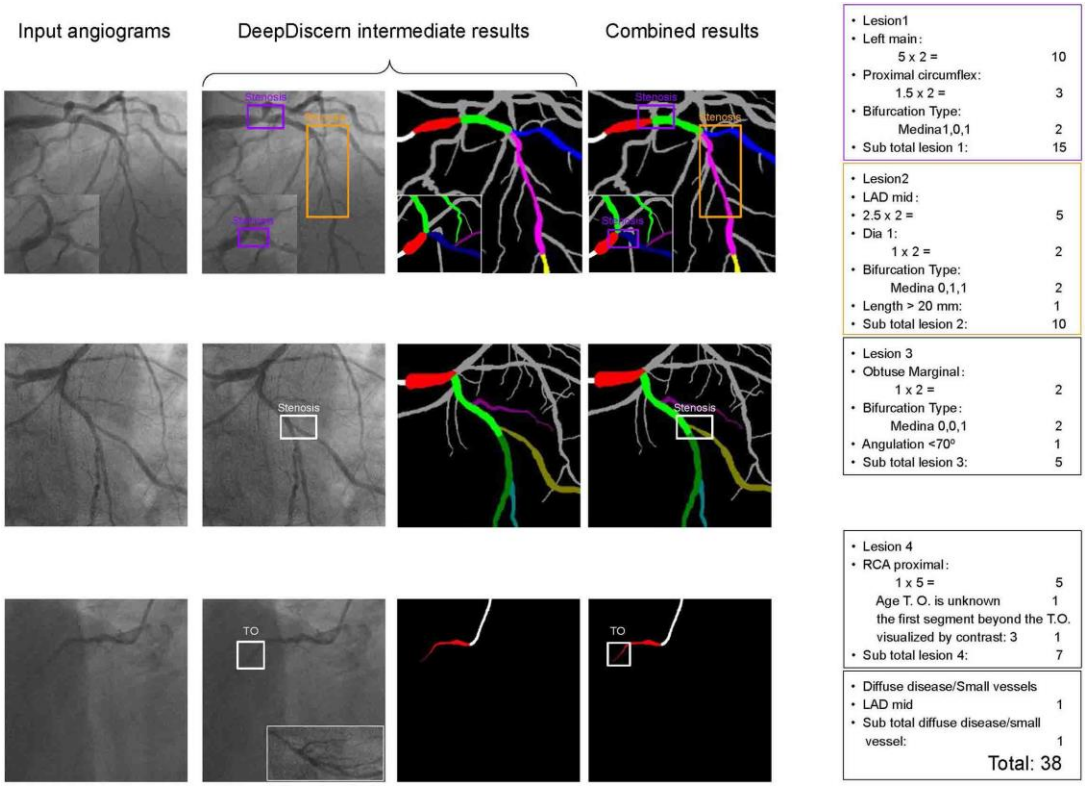
detected lesion cases in the test data set to calculate the precision rate, recall rate and F1 score of every kind of lesion.



Supplementary Figure 6. Performance of vessel extraction.

A) Input angiogram.

B) – F) Result images of recognition models which were trained using a different amount of data (1,000, 3,000, 5,500, 8,000, 11,900 images).



Supplementary Figure 7. Expected automatic calculation system for the SYNTAX score.

First column: input angiograms under different angiographic views. Second and third columns: the intermediate results generated by DeepDiscern system. Fourth column: combined results. All images are zoomed in for better visualisation.

Supplementary Table 1. Coronary arteries labelled in different angiographic views.

Coronary artery segments	CRA[b]	CAU[b]	LAO (right)[a] LAO_CAU (right)[a] LAO_CRA (right)[a]	LAO_ CAU[b]	LAO_ CRA[b]	RAO_CAU[b]	RAO_CRA[b]
LM	√	√		√	√	√	√
LAD proximal	√			√	√		√
First diagonal	√			√	√		√
Add. first diagonal	√			√	√		√
LAD mid	√			√	√		√
Second diagonal	√				√		√
Add. second diagonal	√				√		√
LAD apical	√			√	√		√
Intermediate		√		√		√	
LCX proximal		√		√		√	
LCX distal		√		√		√	
Second obtuse marginal		√		√		√	
First obtuse marginal		√		√		√	
Left posterolateral		√		√		√	
Posterior descending (LCX)		√		√		√	
RCA proximal			√				
RCA mid.			√				
RCA distal			√				
Posterior descending (RCA)			√				
Posterolateral			√				
Unconcerned arteries[c]	√	√	√	√	√	√	√
Background	√	√	√	√	√	√	√
Catheter	√	√	√	√	√	√	√

DIA: diagonal; LAD: left anterior descending artery; LCX: left circumflex artery; LM: left main; L-PDA: left posterior descending; L-PLA: left posterolateral; OM: obtuse marginal; PDA: posterior descending; PLA: posterolateral; RCA: right coronary artery

[a]In the LAO_CAU (right) part of the data set, all angiograms were obtained in the LAO_CAU view. Only the right coronary artery appeared in these angiograms, as well as for LAO and LAO_CRA (right) parts.

[b]In the other parts of the data set, all angiograms were obtained in the angiographic view corresponding to their part name. Only the left coronary artery appeared in these angiograms.

[c]For every part of the data set, all coronary artery segments without check marks were labelled as "unconcerned arteries". These coronary artery segments will not be overly concerned in the angiographic view of this part of the data set.

Supplementary Table 2. National Heart, Lung and Blood Institute (NHLBI) coronary dissection criteria.

Variable	Definition
Dissection	
A	Small radiolucent area within the lumen of the vessel
B	Linear non-persisting extravasation of contrast
C	Extraluminal, persisting extravasation of contrast
D	Spiral – shaped filling defect
E	Persistent luminal defect with delayed anterograde flow
F	Filling defect accompanied by total coronary occlusion

Supplementary Table 3. The mapping relationship between prediction label value and coronary artery segments.

Coronary artery segments	Label value	Colour	Coronary artery segments	Label value	Colour
LM	(255,0,0)		Second obtuse marginal	(128,128,0)	
LAD proximal	(0,255,0)		First obtuse marginal	(128,0,128)	
First diagonal	(0,0,255)		Left posterolateral	(0,128,128)	
Add. first diagonal	(0,128,255)		Posterior descending (LCX)	(128,255,0)	
LAD mid	(255,0,255)		RCA proximal	(255,128,0)	
Second diagonal	(0,255,255)		RCA mid	(255,0,128)	
Add. second diagonal	(128,0,255)		RCA distal	(0,255,128)	
LAD apical	(255,255,0)		Posterior descending (RCA)	(128,128,255)	
Intermediate	(128,0,0)		Posterolateral	(128,255,128)	
LCX proximal	(0,0,128)		Background	(0,0,0)	
LCX distal	(0,128,0)		Catheter	(255,255,255)	

DIA: diagonal; LAD: left anterior descending artery; LCX: left circumflex artery; LM: left main; L-PDA: left posterior descending; L-PLA: left posterolateral; OM: obtuse marginal; PDA: posterior descending; PLA: posterolateral; RCA: right coronary artery

Supplementary Table 4. Recognition performance of all segments under different angiographic views.

Angiographic view	Accuracy % (95% CI)	Sensitivity % (95% CI)	Specificity % (95% CI)	PPV % (95% CI)	NPV % (95% CI)
Left coronary artery					
CRA	98.5 (98.4-98.6)	86.8 (85.8-87.8)	98.9 (98.9-99.0)	74.0 (72.8-75.2)	99.5 (99.5-99.6)
CAU	98.5 (98.5-98.6)	83.8 (82.8-84.8)	98.9 (98.8-99.0)	70.8 (69.7-72.0)	99.5 (99.5-99.6)
LAO_CRA	98.4 (98.3-98.5)	86.6 (85.5-87.6)	98.8 (98.7-98.9)	70.8 (69.7-71.9)	99.5 (99.4-99.5)
LAO_CAU	98.7 (98.6-98.8)	87.8 (86.8-88.9)	99.0 (99.0-99.1)	73.4 (72.1-74.6)	99.6 (99.6-99.7)
RAO_CRA	98.4 (98.3-98.5)	86.6 (85.6-87.5)	98.8 (98.7-98.9)	71.0 (69.9-72.1)	99.5 (99.5-99.5)
RAO_CAU	99.0 (98.9-99.1)	87.8 (86.9-88.8)	99.4 (99.3-99.4)	80.3 (79.2-81.3)	99.6 (99.6-99.7)
Right coronary artery					
LAO	98.1 (98.0-98.2)	78.5 (77.2-79.7)	98.8 (98.7-98.9)	72.8 (71.5-74.2)	99.2 (99.1-99.2)
LAO_CAU	98.6 (98.5-98.6)	83.2 (82.3-84.1)	99.2 (99.1-99.2)	79.7 (78.8-80.6)	99.3 (99.2-99.4)
LAO_CRA	98.1 (97.9-98.2)	86.5 (85.8-87.3)	99.4 (99.3-99.4)	85.1 (84.3-85.9)	99.5 (99.4-99.5)
RAO	97.1 (96.8-97.3)	83.4 (81.7-85.0)	99.2 (99.1-99.3)	79.0 (77.2-80.9)	99.4 (99.3-99.5)

CAU: caudal view; CRA: cranial view; LAO: left anterior oblique view; LAO_CAU: left anterior oblique-caudal view; LAO_CRA: left anterior oblique-cranial view; NPV: negative predictive value; PPV: positive predictive value; RAO: right anterior oblique view; RAO_CAU: right anterior oblique-caudal view; RAO_CRA: right anterior oblique-cranial view