Technical details of automatic self-supervised feature extraction

Data pre-processing

Assume $X=(x_1,x_2,\ldots,x_m)$ denotes one input video sample with m frames, where x_i is the i^{th} frame. For an input video frame x, we first randomly crop a sub-area, and then transform them into z_1 and z_2 by different data transformations:

$$z_k=T(x),k=1,2 \quad [1]$$

where T() includes random color distort and Gaussian blur. After data transformation, each z_k is divided into 3×3=9 tiles while leaving a gap (about 6 pixels) between two adjacent tiles as $z_k=\left\{z_{k_1},z_{k_2},\cdots,z_{k_9}\right\}$

Network architecture

A Siamese network with 9 (which is the number of tiles) sharing weight branches is adopted to solve the proxy task. The backbone network φ is 2D ResNet-34 excluding the last fully-connected layer. We can obtain feature representation as:

$$f_{k_j}=\varphi\left(z_{k_j}\right), j=1,2,\cdots,9;k=1,2 \quad [2]$$

Structure recovery

We formulate a proxy task which aims to rearrange and recover the structure. We first yield all the permutations (P) of tiles, i.e., $P=(p_1,p_2,\ldots,p_{9!})$ and iteratively select H ($H\leq 9!$) permutations with the largest Hamming distance from P, i.e., $P^\wedge{}'=(p_1,p_2,\ldots,p_H)$. Then the 9 tiles of z_k are rearranged according to a random selected p from permutation pool P'. Therefore, the network is trained to identify the selected permutation. The feature f_k' can be obtained by feature concatenation of $\left(f_{k_1},f_{k_2},\cdots,f_{k_9}\right)$, then the predicted possibilities l of each permutation can be generated via:

$$l=-g\left(f_k'\right) \quad [3]$$

where g represents a fully-connected layer. Assume the index of chosen permutation for each z_k is y, the loss (L_{sr}) can be defined as:

$$Lsr=-\sum_{i=1}^{H}y_i\log l_i=\sum_{k=1}^{2}\sum_{i=1}^{H}y_{ki}\log l_{k_i} \quad [4]$$

Color transform toleration

We design another proxy task to force the network more concentrate on color-correlated information. Assume a subset $\{x\}$, which may belong to different videos, is sampled in each mini-batch, the feature representations in each mini-batch are $\{f_{ik_j}$; $i=1,2\ldots,N,k=1,2$; $j=1,2,\ldots,9\}$, where N is the size of mini-batch. The f generated from the same x is regarded as a positive pair, and vice versa. The network is force to minimize the difference between positive pairs and enlarge the negative ones.

$$L_c=-\log\sum_{i=1}^{N}\sum_{j=1}^{9}\frac{c\left(f_{i1_j},f_{i2_j}\right)}{\sum_{p=1,p\neq i,k'=k''=1,2}^{N}c\left(f_{pk_j'},f_{pk_j''}\right)} \quad [5]$$

where $C(x,y)=\exp\left(\frac{x^Ty}{\tau\|x\|\|y\|}\right)$, and τ is a temperature parameter.

Objective

Our total loss function of our SSL feature extraction can be defined as:

$$L=L_{sr}+L_c \quad [6]$$

MR jet recognition and segmentation

Feature encoding

Our backbone model φ is then transferred to downstream tasks, namely MR jet recognition task and segmentation task (shown in *Figure 2B*). Since X may consist of several cardiac cycles, we let $E=(e_1,e_2,\ldots,e_m)$ denotes a one-hot ground truth indicating the max MR jet area frame, and $Y=(y_1,y_2,\ldots,y_m)$ denotes the segmentation ground truth. The segmentation ground truths of those desirable frames are acquired, where e_i=1, and e_i=0 vice versa. We first crop a central area of each frame and then obtain feature representations via:

$$f_i=\varphi(x_i),i=1,2,\ldots,m \quad [7]$$

The max MR frame recognition

The $\{f_i\}$ are then concatenated into f' along the time dimension. A 3D decoder D_r, which consists of two 3D convolution layer, one 2D pooling layer, and one fully-connected layer, is employed to generate predicted label $E'=\{e_1',e_2',\ldots,e_m'\}$. The loss function is represented as:

$$L_r=\|E'-E\|^2=\|D_r\left(f'\right)-E\|^2=\sum_{i=1}^{m}\|e_i'-e_i\|^2 \quad [8]$$

The max MR frame segmentation

We integrate the information of those previous frames, which lack of segmentation ground truth, by introducing the long short-term memory (LSTM) architecture to explicitly promote the exploring of all video frames for better segmentation

https://dx.doi.org/10.21037/atm-21-3449

reconstruction. Assume f_k is one of the max MR frames. Then the integrated feature is:

$f_k'=LSTM(f_1,f_2,\ldots,f_{k-1},f_k)$ [9]

Then f_k' is fed into a 2D decoder D_s with skip-connection to obtain predicted segmentation y_k'. Segmentation loss L_s is generated via dice loss.

$$L_s=\sum_{i=1}^{m} I_{e_i\neq 0} Dice\left(y_i',y_i\right)=\sum_{i=1}^{m} I_{e_i\neq} Dice\left(D_s\left(f_k'\right),y_i\right) \quad [10]$$

where I is an indicator function evaluating to 1 if e_i≠0, and vice versa.

Objective

Our total objective of multi-task framework is:

$L=L_r+L_s$ [11]