

Frame Classification manual

(version 1.0)

GICI group

Department of Information and Communications Engineering

Universitat Autònoma Barcelona

<http://www.gici.uab.es> - <http://gici.uab.cat/GiciWebPage/downloads.php>

January 12, 2015

1 Description

This source codes are the implementation of frame classification for X-ray coronary angiograms.

Normally one X-ray coronary angiogram presents four phases of the contrast agent flow motion: pre-perfusion, when only the catheter is visible; inflow, when the contrast agent is injected into the coronary vessels but has not yet reached all the vessel branches; complete state, when the entire vessel tree is visible; and washout, when the perfusion is finished and the contrast agent starts leaving the coronary arteries. Example frames for these four phases from one coronary angiogram are presented in Fig. 1.

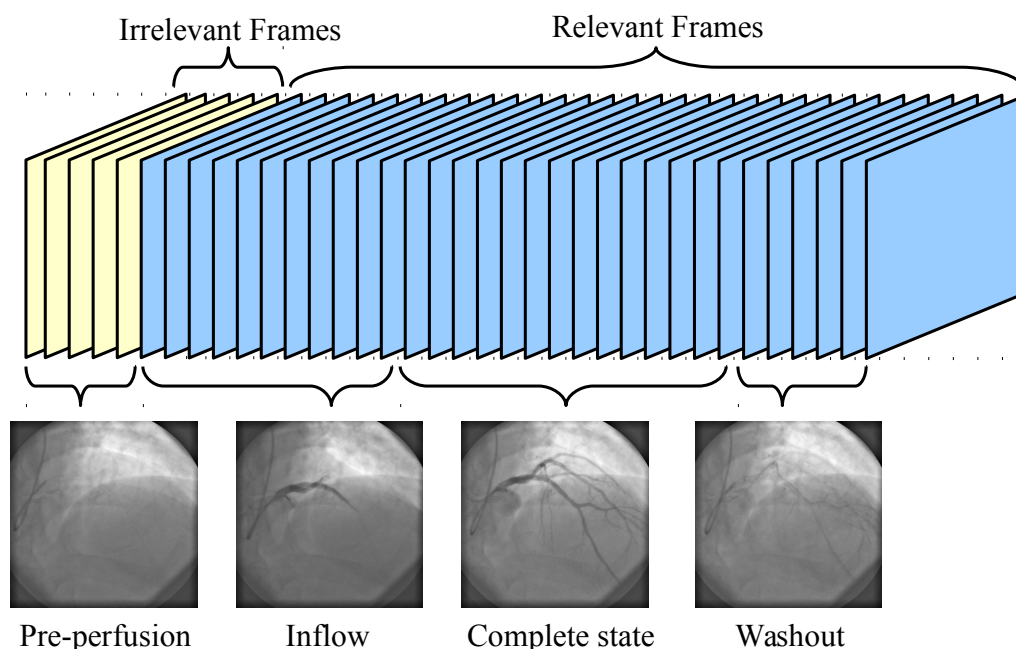


Fig. 1: Coronary angiogram sequence showing the irrelevant and relevant frames and the 4 phases of the contrast agent flow.

Our frame classification program can identify those frames belonging to the pre-perfusion phase, which are clinical-irrelevant. And a selective lossy + lossless compression can be employed based on the classification results for coronary angiograms, in order to improve the compression performance.

The following functions are included:

SOLO_FRANGI which applies the Frangi filter [1] onto the coronary angiogram frame by frame.

FRAME_CLASSIFICATION which gives the frame number of the last irrelevant frame in the coronary angiogram.

DEMO which is a demo program to do frame classification on an example coronary angiogram in the package.

2 Requirments

This software is programmed in MATLAB, so you might need a MATLAB software from MathWorks to run this application. We have used MATLAB 2013a.

3 Usage

The example of using function **SOLO_FRANGI** and **FRAME_CLASSIFICATION** is shown below. You can run the following command line by line in the command window of MATLAB:

```
>> close all

% add path
>> addpath(genpath('./'))

% read image file and do frangi enhancement
>> t = cputime; % count the running time
>> file_name = '03520798-80x512x512-15.000-2_Bytes-LittleEndian';
>> ln = 80;
>> solo_frangi([file_name, '.raw'], ln, file_name);

% compute the frame number of pre-perfusion phase
>> LN1 = frame_classification([file_name, 'after_frangi.raw'], ln);

% remove the enhanced image file
>> delete([file_name, 'after_frangi.raw']);

% display the running time
>> t = cputime - t;
>> fprintf('The time for computing is %d seconds\n', t);
```

Or you can creat one .m file in the editor of MATLAB and run it once in the command window of MATLAB, which is done in file 'demo.m', and only the following command is needed:

```
>> demo
```

More details about choosing parameters inside each function, please refer to the notes in the .m files in the package.

4 Notes

If you need further assistance, you might want to contact us directly.

References

- [1] A. F. Frangi, W. J. Niessen, K. L. Vincken, and M. A. Viergever, "Multiscale vessel enhancement filtering," in *Medical Image Computing and Computer-Assisted InterventionMICCAI98*. Springer, 1998, pp. 130–137.