Norbert Health tests

Oussama Ennafii

June 15, 2021

Computer Vision

1

In order to register the two images, one can make use of a standard procedure in 3D vision for finding a rigid transformation (RANSAC) from one image to the other. This supposes that both images do not suffer from distortion. This process relies on three steps:

1. Detect invariant interest points based on efficient methods such as SIFT, SURF or ORB.
2. Match the interest points based on their descriptors.
3. Use a RANSAC derived method to find the corresponding homography which will be robust to outlier matches.

This is described in details in Alg. 1.

```
Data: lwir_image, gray_image, interest_point_type ∈ {SIFT, SURF, ORB},
      descriptor_distance, min_matches
Result: Registered LWIR image.
lwir_points, lwir_descriptors = compute_interest_points(lwir_image, interest_point_type);
gray_points, gray_descriptors = compute_interest_points(gray_image, interest_point_type);
matches = knn_matches(lwir_descriptors, gray_descriptors, descriptor_distance);
fitered_matches = lowe_ratio_test(matches, ratio = 0.7);
if |fitered_matches| < min_matches then
    raise runtime_error("Insufficient matches to compute registration");
end
H = compute_homography(lwir_points, gray_points, matches);
registered_lwir_image = apply_homography(lwir_image, H);
```

Algorithm 1: RANSAC based registration.

Since both images do not have the same domain the descriptors may not be well suited in this case. [2] proposes the use a restricted SURF or SIFT in scale to alleviate this problem. In order to solve this problem of descriptor domain mismatch, other approaches could be used. One way to do so would be to learn a useful distance function between these descriptors. Another way would rely on extracting edges (with Canny Edge detector for instance) and trying to match the corresponding edges (with Iterative Closest Point (ICP) for example).

LWIR images once registered could be used to compute a thermal (heat) map for a patient for instance.

2

Check folder xxx

Audio coughing detection

Since the intraclass variance of the `breathing` class is very high compared to `coughing` class, the first thing to be done is to consider a mutliclass classification instead of a binary one. These classes would be: `coughing`, `normal breathing`, `wheezing`, `snoring`, `gasping`, `panting` and `snorting`. There are in total seven classes. For the moment classes are supposed to be exclusive (i.e. not a multilabel problem) if we suppose that observations are small in length.

We suppose that all audio files have the same number of channels, the same sampling resolution and the same length. Different approaches could be used. Each one is detailed in the following sectionss.

The dataset should be divided to two sets. A validation set containing a representative amount of each class, containing ideally 10% of signals in total for instance. The goal of such a set is to control the performance of each approach. The rest is used for training using a 10-fold stratified cross validation approach in order to calibrate the parameters of the used models.

Naive approach

Use the Mel spectrogram to respresent the signal. The latter relies on respresenting the signal by a windowed fourier transform with frequencies mapped to the Mel scale. In order to reduce the number of coefficients, one would compute a histogram for each Mel frequency across time. The resulting matrix ($number\_of\_frequencies \times number\_of\_bins$) will be flattened to obtain a feature vector per audio signal. This feature vector will be fed to a supervised classifier such as SVM or Random Forest.

Scattering transform

The scattering transform could be seen as a multiscale generalization of the Mel spectrogram [1]. As such, instead of computing the Mel spectrogram as in the previous approach, we can compute the scattering transform of each signal. This will result in a cascade of coefficients that could be reduced using once again a histogram. The resulting feature vector is then fed to a classifier as before.

CNN on Mel spectrogram

The Mel spectrogram is seen here as a single channel image. Based on this observation, one way of computing better features would be to train a single channel CNN with a cross entropy loss. The learned filters would look at neighboring times and frequencies for short range (in time and in frequencies) patterns in the signal.

Naive LSTM

Another approach would be the use the Recurrent neural networks for classifying the audio signals. Since the whole signal is classified, a long range in time is needed. That is why Long Short Term Memory (LSTM) networks seem like a good choice. In this case, the LSTM is applied to the raw signal. This is not supposed to work well but should be tried nonetheless as a second baseline.

Mel LSTM

Instead of applying the LSTM directly on the signal, first the Mel spectrogram is computed. For eah time the network is fed the frequency amplitudes as features. This is expected to work best for this problem.

References

[1] Joakim Andén and Stéphane Mallat. Multiscale scattering for audio classification. In *ISMIR*, pages 657–662. Miami, FL, 2011.

[2] Chunlei Huo, Chunhong Pan, Leigang Huo, and Zhixin Zhou. Multilevel sift matching for large-size vhr image registration. *IEEE Geoscience and Remote Sensing Letters*, 9(2):171–175, 2011.