ISSN: 2096-3246
Volume 54, Issue 01, Jan, 2022

https://doi.org/10.5281/zenodo.12707384

A NOVEL APPROACH FOR HUMAN ACTIVITY RECOGNITION FOR ANDROID

ASONDI SREEPRADHA,VIJAYA BHASKAR MADGULA,TALARI SIVALAKSHMI
Assistant Professor[1,2,3],
sreepradha5217@gmail.com,vijaya.bhaskar2010@gmail.com,,shivalakshmidinesh@gmail.com
department of CSE, Sri Venkateswara Institute of Technology,
N.H 44, Hampapuram, Rapthadu, Anantapuramu, Andhra Pradesh 515722

Keywords:

human action recognition , real world problems

ABSTRACT

With the recent advancements in computer vision, Human Action Recognition has become one of the aesthetic research fields. There has been little research work in the real world conditions of human action recognition systems, which encourages us to seriously search in this application domain. Although lot of healthy approaches have been introduced in the literature, they are still insufficient to fully cover the challenges. This review highlights the advances of state-of-the-art activity recognition approaches, especially for the activity recognition and classification methods. For the representation methods, we sort out a chronological research trajectory from global representations to local representations, and recent depth-based representations. For the classification methods, we conform to the categorization of template-based methods, discriminative models, and generative models and review several prevalent methods.

A ES
ISSN: 2096-3246
Volume 54, Issue 01, Jan, 2022
https://doi.org/10.5281/zenodo.12707384

Introduction

Human activity recognition on mobile devices is a rapidly growing field in computer vision community as it is involved in the development of many important applications such as human computer interaction, virtual reality, security, video surveillance and home monitoring.Human activity recognition systems implementation is done in 2 methods:

1) Contact-based.

2) Remote methods.

Contact-based systems require the physical interaction of the user. So these methods are impacted by the nature and source of data (example: accelerometers, wearable sensors etc). Since the sensors need skills and a lot of equipment in order to implement real world applications these sensors should be available easily and user must be acceptable to perform continuous tasks. To simplify the task of human computer interaction the remote methods are employed.
The remote methods (like vision based) allows user to convey their ideas through gestures. Since these vision based systems uses images or video to recognize activities, vision based systems does not require users to wear sensors which provides an edge over other approaches.

Humanactivity recognition also makes it possible to develop mobile health applications with some implementation that track a user's activities. The work described in this paper relies on Android smartphones.In this paper we employ a straight-forward approach for human activity recognition. We collect images of the users actions into labelled data and then we induce a model by applyingalgorithms to the training examples.

II. PROPOSED ALGORITHM

Human activity recognition from a single image consists of image insertion, model building and then testing. The model evaluation is done manually by the developer.

- Convert the video into frames.
- Pre-process image by converting YUV to RGB format.
- Convert RGB to bitmap image.
- Bitmap image is classified by classifier.
- Model inference makes predictions on the data received from classifier and stores it in a buffer.
- Image analyzer unwrap the confidence score from buffer.
- Application displays the output.

Converting YUV to RGB:

YUV is a color encoding system typically used as part of a color image pipeline. It encodes a color image or video taking human perception into account, allowing reduced bandwidth for chrominance components, thereby typically enabling transmission errors or compression artifacts to be more efficiently masked by the human perception than using a "direct" RGB-representation.[1]

Generally, RGB files areencoded in 8, 12, 16 or 24 bits per pixel andYUV files are encoded in 12, 16 or 24

ISSN: 2096-3246
Volume 54, Issue 01, Jan, 2022

https://doi.org/10.5281/zenodo.12707384

bits per pixel. The common formats are YUV444 (consists of 12 bytes for 4 pixels), YUV411 (consists of 8 bytes for 4 pixels), YUV422 (consists of 6 bytes for 4 pixels) and YUV420 (consists of 6 bytes for 4 pixels in reordered format). Y is a measure of overall brightness or luminance. U and V are computed as scaled differences between Y and the blue and red values. To convert from YUV to RGB and vice- versa, it is simplest to use RGB888 and YUV444.

Conversion in code:

```
void YUVImage::yuv2rgb(int y, int u, int v, int *r, int *g, int *b)

  const { int red = y + (351*(v-128))>>8;

  int green = y - (179*(v-128) + 86*(u-

  128))>>8; int blue = y + (443*(u-128))>>8;

  *r = clamp(red, 0, 255);

  *g = clamp(green, 0, 255);

  *b = clamp(blue, 0, 255);

}
```

Bitmap:

Bitmaps are the array of binary values representing the values of pixels in an image or display. It is a method by which a display space is defined, including the color of each of its pixels.Bitmaps are perfect for creating detailed images (like photographs) because of the amount of data each pixel can store. The greater the amount of data, the broader the range of colors it can display. And, it's much easier to create a colorful image with realistic, transitioning color tones when you have access to a full range of colors.

Classifier and model inference:

During training, an image classification model is fed images and their associated labels. Each label is the name of a distinct concept, or class, that the model will learn to recognize.Given sufficient training data (often hundreds or thousands of images per label), an image classification model can learn to predict whether new images belong to any of the classes it has been trained on. This process of prediction is called inference.

Confidence score:

In learning algorithm, Confidence defines the probability of the event (or probability of input to fall in different classes). Confidence value can be calculated for single input as well giving the meaning as how much the algorithm is confident for that class.

Data Processing:

- All the collected images should be in same folder.
- Split the data as a directory of sub-directories depending on type of image.
- Each sub-directory is named as label for the images contained in it.

ISSN: 2096-3246
Volume 54, Issue 01, Jan, 2022
https://doi.org/10.5281/zenodo.12707384

```
|
|-- human_poses
|    |
|    |-- standing
|    |    |-- image_1.jpg
|    |    |-- image_2.jpg
|    |    |-- ...
|    |
|    |-- sitting
|    |    |-- image_1.jpg
|    |    |-- image_2.jpg
|    |    |-- image_3.jpg
|    |    |-- ...
|    |    |
|    .
|    .
|    .
```

III.EXPERIMENTRESULTS

Selecting the appropriate testing image is the most important process in the project.Single Image from testing dataset is taken such a way that it reaches our implementation of amodel. Each image from the test dataset is taken as an input and process it further toproduce an output. This selection of image is more important because implementation ofmodel requires an image input for processing.When you subsequently provide aimage as input to the model, it will output the probabilities of the image representing each of the types of actions it was trained on.

An example output might be as follows:

ISSN: 2096-3246
Volume 54, Issue 01, Jan, 2022

https://doi.org/10.5281/zenodo.12707384

Action	Probability
Sleeping	0.01
Drinking	0.03
Sitting	0.02
Standing	0.94

Each number in the output corresponds to a label in the training data. Associating the output with the three labels the model was trained on, you can see that the model has predicted a high probability that the image represents a standing.

ISSN: 2096-3246
Volume 54, Issue 01, Jan, 2022
https://doi.org/10.5281/zenodo.12707384

Ambiguous results:

Since the output probabilities will always sum to 1, if an image is not confidently recognized as belonging to any of the classes the model was trained on you may see the probability distributed throughout the labels without any one value being significantly larger.

For example, the following image might indicate an ambiguous result:

Action	**probability**
Sleeping	0.06
Drinking	0.49
Sitting	0.04
Standing	0.41

III.CONCLUSION

In this paper we describe and evaluate a vision basedapproach for implementing activity recognition, usingonly smartphones. We demonstrate that nearly perfectresults can be achieved if a personalized

ISSN: 2096-3246
Volume 55, Issue 02, Nov, 2023

https://doi.org/10.5281/zenodo.12707384

model isconstructed, even using only a very small amountof user-specific training data. We further show thatimpersonal models perform much worse than personalmodels, even under the best conditions where they aretrained on similar users. Analysis of the data showsthat impersonal models cannot effectively distinguishbetween certain activities, whereas personal modelscan effectively learn the user-specific differences thatconfound impersonal models. We also show that whilethe poor performance of impersonal models is affectedby some idiosyncratic users, whose activities cannot beaccurately predicted, the problem is widespread andnot restricted to a few problem users. Although thereappears to be room for marginal improvement in theaccuracy of impersonal models by increasing the training data.

REFERENCES

1. https://en.wikipedia.org/wiki/YUV
2. Human activity recognition and Behaviour analysis- For cyber physical Systems in smart environment – Chen, Liming, Nugent, ChrisD.
3. Human activity recognition: using wearable Sensors and smartphones(Champman& Hall/CRC computer and information science series book 30).
4. Advanced Applied Deep Learning: Convolutional Neural Networks and Object Detection(Umberto Michelucci)
5. https://ieeexplore.ieee.org/document/7072750
6. https://www.sciencedirect.com/science/article/abs/pii/S0031320315000953

https://www.microsoft.com/en-us/research/project/human-action-and-activity-recognition/