Proposal of Gesture temporal detection pipeline for news videos for the GSOC 2021 at RedHenLab

Yunfei ZHAO

October 27, 2021

Summary of the Proposal

Gesture recognition becomes popular in recent years since it can play an essential role in many fields, such as non-verbal communication, emotion analysis, human-computer interaction, etc. We can notice that people use quite a lot of hand gestures in daily life. The research task is to detect hand gestures in raw news videos that are streams of RGB images. I propose a keypoints-based pose tracking system for human tracking and a Transformer and keypoints-based gesture detector for gesture detection to fulfill this task. This structure is composed of a keypoints extractor, a person tracker, and a gesture detector. So the mission has three main parts. The first part is to track people in temporal space. In the second part, for each person, we use their hand keypoints features in temporal space to construct several keypoints sequences. The third part is to use these sequences to make predictions of the existence of gestures. I believe that for gesture detection tasks, both spatial and temporal information is important. So that is why we use the Transformer that can take into account the local shape information of hands in one frame and can also capture the global hand motion information across the frames. As hand gestures in news videos do not have a good definition of label class, we start with only detecting the existence of a hand gesture. The classification can be easily extended. The final evaluation will be done on Redhen's "Newscape Dataset".

Background

Available resources

Thanks to Redhen community, we have been provided with several interesting resources and useful tools. I collect also some information which may help us during the realization of our project. These resources build a very good starting point and I will list them below:

- Openpose [1] the first real-time multi-person system to jointly detect human body, hand, facial, and foot keypoints (in total 135 keypoints) on single images. It is one of the state-of-art for human keypoints detection with very robust implementation. Using bottom-up approach, it has outstanding performance in multi-person scenario, both in accuracy and speed in compare with Mask-Rcnn and Alpha pose. Convolutional Pose Machines [2] is an interesting human pose estimation method as well.
- Hand Keypoint Detection in Single Images using Multiview Bootstrapping [3], this paper deals with the hand key point detection by a Bootstrapping method. The main ideal is to using multiview of the same of scene to solve the osculation problem of hand keypoints. It have been integrated in the OpenOpen hand keypoints detection module and it will help us to do the hand keypoints and features extraction. We possess also a tutorial [4] for using it in Opencv.
- Attention Is All you Need[8] [5]. This article introduces our famous Transformer model which is the most popular model for attention mechanism. It is firstly used in NLP task.
- 15 Keypoints Is All You Need[6] [6], a new paper published in early 2020. They use body keypoints and the transfer to do pose tracking for human.

[1] https://github.com/CMU-Perceptual-Computing-Lab/openpose

[2] https://arxiv.org/abs/1602.00134

[3] https://arxiv.org/abs/1704.07809

[4] https://learnopencv.com/hand-keypoint-detection-using-deep-learning-and-opencv/

[5] https://arxiv.org/pdf/1706.03762.pdf

[6] https://arxiv.org/abs/1912.02323

- Simple Baselines for Human Pose Estimation and Tracking[10] [7], a multi-persons human pose and tracking model published by microsoft in 2018.

- End-to-End Object Detection with Transformer[1] [8]. In 2020, Facebook AI published "End-to-End Object detection with Transformers" that apply the transformer in object detection tasks and set a new baseline. In this article they present a new method that view object detection as a direct set prediction problem and apply transformer in object detection task.

- Temporal Action Localization in Untrimmed Videos via Multi-stage CNNs[4] [9], this paper introduced a pipeline to localise temporal action localization in untrimmed videos by generating varied length segments of a video and than do a classification followed by a post-processing to select the best prediction. For a segment, it takes into account both predicted class probability and the overlap with their ground truth.

- Sergiy Turchyn developped a project named A Visual Search Engine for Gesture Annotation. [10].They using skin color and optical flow to extract the body part motion to detect body parts. But the main drawback for this method is the execution time. For hand motion detection, it takes 50 minutes per image.

- During GSOC 2019, Abhinav Patel contribute a project for gesture detection and recognition in TV News Videos[11], He transforms the video problem as a simple image problem, but the performance of his pipeline is not evaluated. His contribution can be helpful for this year's work.

- C3D, Learning Spatiotemporal features with 3D Convolutional Networks[7] attributed by Facebook AI Research, that use 3D convolutional neural network to extract features from videos. And this is the structure which I am going to use to build our gestures detector.

- Okan Köpüklü et al. proposed a hierarchical structure for gesture detector and classification in Real-time Gesture Detection and Classification Using Convolutional Neural Networks [2] and the structure I proposed is also inspired by their structure.There is also a structure proposed by Guangming Zhu et al. in Multimodeal Gesture Recognitio Using 3-D Convolution and Convolutional LSTM [12], which may be considered in our project.

Challenges

In this part, I am listing some difficulties we may face during the project. These challenges are the main problem we are going to solve during the 3 months work.

1. In the video stream, We need extract hand information for each person. If there is a gesture, it is performed by one person normally, so we need to collect the information for this gesture from several frames.

2. In spacial space, we need to locate a gesture in a frame. In temporal space, we need to locate the begin and the end of a gesture. We need also to classify a gesture.

3. Given a raw RGB image of gesture we obtain from the original image, we need a stable model to extract hand keypoints. As occlusion in hands is a fatal problem, we need to find a efficient resolution. For example, in OpenPose, I notice than not all keypoints can be detected during a gesture. So we need to come up with a solution to improve it as a lost of keypoint can cause a degradation during the following process and it will have a bad influence on our recall rate.

4. Run-time efficiency need to be considered too as we will run our model on big dataset. For example the UCLA library broadcast NewsScapes dataset contains more than 400,000 news programs. So our model need to be run close to real-time.

[7] https://openaccess.thecvf.com/content_ECCV_2018/papers/Bin_Xiao_Simple_Baselines_for_ECCV_2018_paper.pdf

[8] https://www.ecva.net/papers/eccv_2020/papers_ECCV/papers/123460205.pdf

[9] https://arxiv.org/abs/1601.02129

[10] https://www.redhenlab.org/home/the-cognitive-core-research-topics-in-red-hen/video-processing-pipeline/gesture-detection-2017

[11] https://github.com/abhinavpatel2912/Gesture-Detection

Goal and Objectives

1. Build a real-time and reliable tracking system for human.

   Objectives:

   1, This tracking system is the base of our model, since the detection process is based on it. We need to detect people and track them with a very robust model. The detector need to have a very high recall rate. because if we fail to track the person, we can not collect his or her hand information.

   2, We need also have a reliable model to extract hand information from a tracked person. It includes hand keypoints, hand location in a frame and their performer.

2. Build a robust hand gesture detector.

   Objectives:

   1, We need to locate the begin and the end timestamp of a gesture on a video

   2, The detector need to have high precision and recall rate.

3. Implement the whole pipeline.

   Objectives:

   1, The whole pipeline will located within a Singularity module that is will be installed on Red Hen's high-performance computing clusters and will be tested on different datasets.

   2, The whole pipeline should have an acceptable run speed.

Methods

Output example

Example of output for one frame

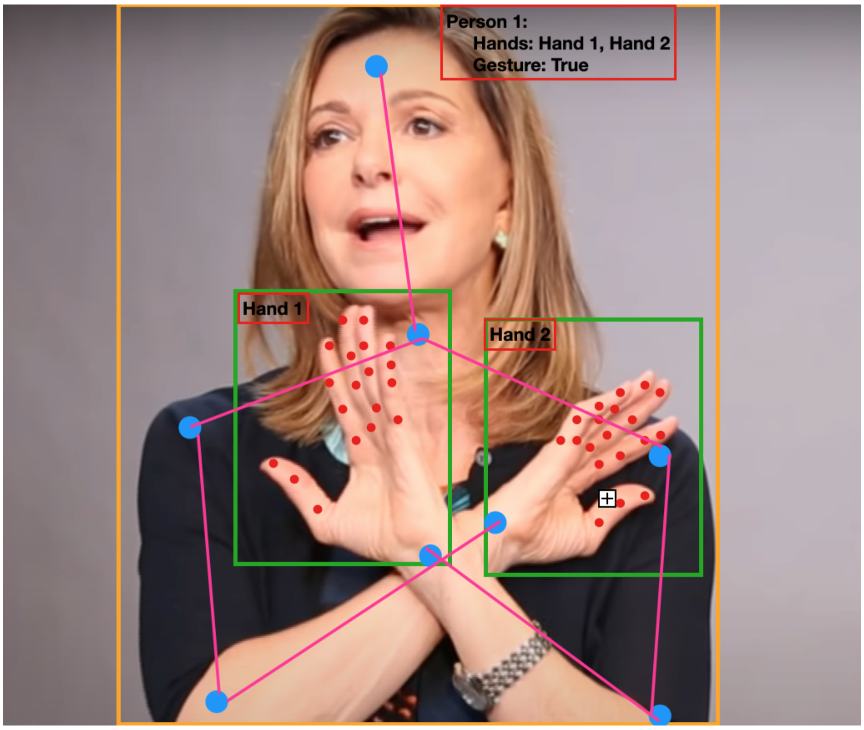


Figure 1: An example of output result for one frame where a gesture is performed

From figure 1, We can have a clear demonstration of the output for a frame in a sequence of frame where a gesture is detected. More detailed information can be demonstrated, such as the confidence of each possible gesture, the duration of the gesture, the timestamp in the video, etc.

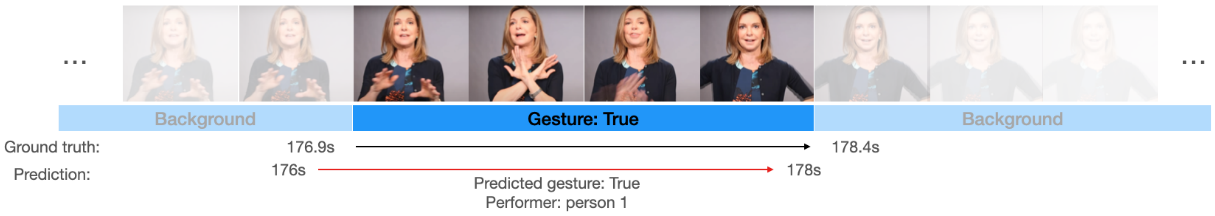


Figure 2: An example of output result for a video segment where a gesture is performed.

Example of output for one video

From figure 2, We can have a clear demonstration of the output for a video segment where a gesture is detected.

Pipeline

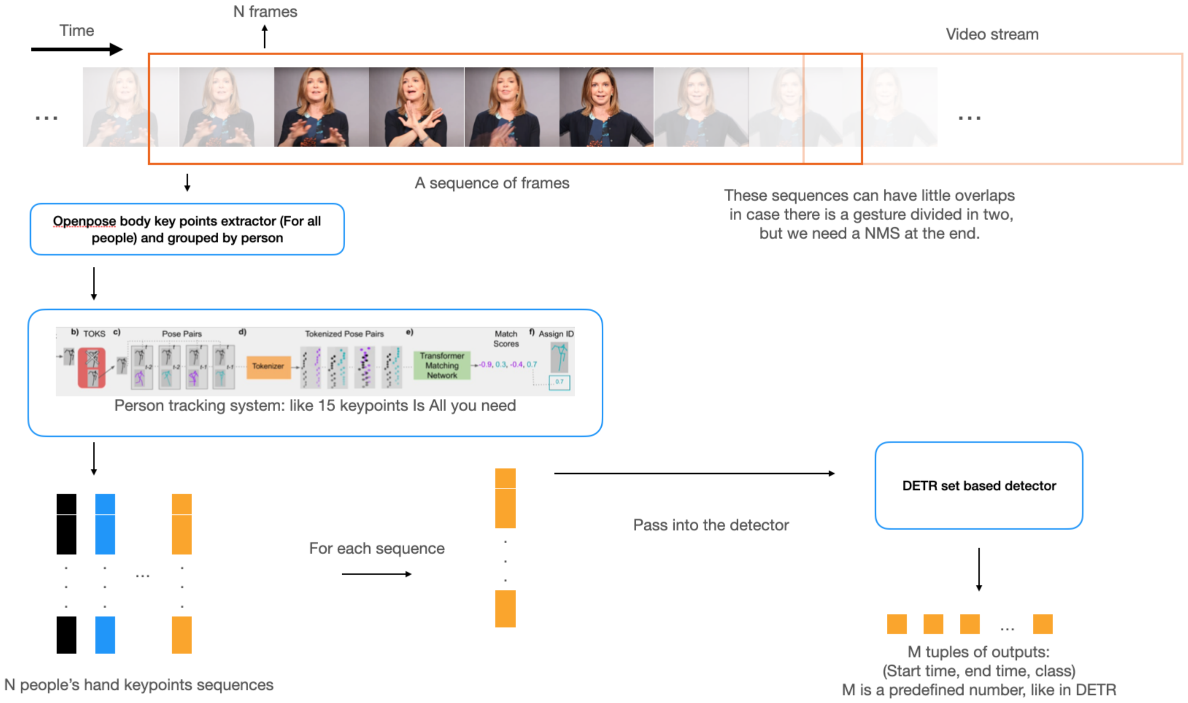

Time
N frames
Video stream
...
...
A sequence of frames
These sequences can have little overlaps
in case there is a gesture divided in two,
but we need a NMS at the end.
Openpose body key points extractor (For all people) and grouped by person
b) TOKS
c)
Pose Pairs
d)
Tokenized Pose Pairs
e)
Match Scores
f) Assign ID
Tokenizer
Transformer Matching Network
-0.9, 0.3, -0.4, 0.7
0.7
Person tracking system: like 15 keypoints Is All you need
...
For each sequence
Pass into the detector
DETR set based detector
...
N people's hand keypoints sequences
M tuples of outputs:
(Start time, end time, class)
M is a predefined number, like in DETR

Figure 3: The pipeline for gesture detection given a video stream. We extract sequences of frames each time from the stream. We use Openpose to extract keypoints for all people. We use a human pose tracking system based on keypoints and the Transformer to track each person to get their keypoints sequence. For each keypoints sequence we apply our transformer detector to locate gestures

Ideas for foreseeable challenges

In this part, I will present some ideas to tackle the challenges I list in the former section. These ideas will be discussed and will be improved.

- **Challenge 1**: Firstly, we need to extract all keypoints for each frame by Openpose, than following the idea of "15 Keypoints is All You Need"[6] or "Simple Baselines for Human Pose Estimation and Tracking"[10]. The implementation of the later is available on github and it is also meaningful to contribute a similar pose tracking system for Openpose. For each track, we store their hand key points to generate a hand keypoints stream and we need also store the corresponding person bounding box and their hand position based on the hand keypoint module in openpose.
- **Challenge 2**: In "End-to-End Object Detection with Transformer"[1], they present a new method that view object detection as a direct set prediction problem. DETR predicts all objects at once, and

is trained end-to-end. DETR does not have multiple hand-designed components that encode prior knowledge, like sliding window, spatial anchors, non-maximal suppression, large set of proposals, or window centers. Inspired by this model, we can treat our gesture detection task from hand keypoints stream as a direct set prediction too, as number of gestures in a video is not enormous, so we can for example suppose that we have maximum 10 gestures during 1 minutes. And we predict directly the start, the end of these gestures directly from a sequence of a keypoints stream. We will of course have $\varnothing$ for no gesture.

- **Challenge 3**:To have a robust **hand ketpoints extractor**. OpenPose follows the idea of this paper [5]. But from my experience, I found that there is fatal **hand keypoints losing problem** due to the occulation. First idea is to use more labeled data to improve the model precise. **Second idea** is to regress the losing keypoint from the known keypoint or from the adjacent video frame. This is a very interesting reseach point as there is no published method to tackle this problem and we can also improve the hand keypoints module for OpenPose.

- **Challenge 4**: As OpenPose is the fast model we have, especially in multi-person scene, So the bottleneck is at the tracking system and transformer detector. But this bottleneck may be mitigated by parallel computing.

In conclusion, the whole pipeline takes into a raw RGB video. We divide the video in sequences of frames. A sequence of frame is passed into OpenPose body keypoints extraction process one by one and after that for each sequence, each person is tracked and we extract hand keypoints for each person to generate hand keypoints sequences, subsequently. Finally, the sub-sequences are passed to our transfer detector for gesture detection. The output is showed in figure 1.

Our project can be used to do the gesture detection and recognition in videos. As it is designed for multi persons and real time, it can be expanded to a voting system, for example, in a classroom, when we what to count the votes , we can ask each student to perform a gesture corresponding to the choice they made, then we can count the votes by counting the gestures. It can also be used for multi persons video games which need gesture interaction.

Available dataset

- **Jester Dataset**[3][12]: The 20BN-JESTER dataset is a large collection of labeled video clips that show humans performing pre-definded hand gestures in front of a laptop camera or webcam.

- **EgoGesture Dataset**[11][13]: EgoGesture is a multi-modal large scale dataset for egocentric hand gesture recognition. This dataset provides the test-bed not only for gesture classification in segmented data but also for gesture detection in continuous data.The dataset contains 2,081 RGB-D videos.

- **Chalearn LAP ConGD Database**[9][14]: ChaLearn LAP Isolated Gesture Dataset (IsoGD) and ChaLearn LAP Continous Gesture Dataset (ConGD). There are gesture dataset of RGB or RGB-D videos.

- **DVS128 Gesture Dataset**[15]: The dataset that was used to build a real-time, gesture recognition system described in the CVPR 2017 paper titled "A Low Power, Fully Event-Based Gesture Recognition System."

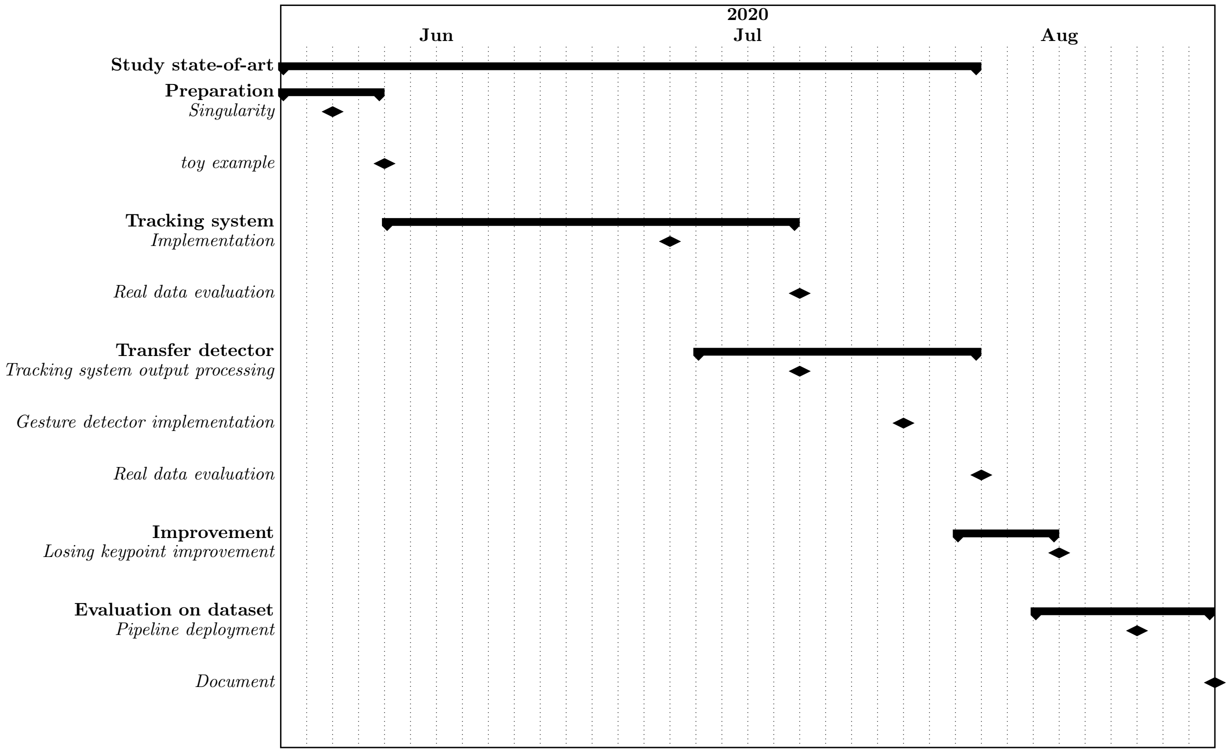
- **The UCLA Library Broadcast NewsScape dateset**[16]:NewsScape, which contains more than 400,000 news programs from the United States and around the world from 2005 to the present. And this is probably the dataset we are going to test our pipeline.

[12] https://20bn.com/datasets/jester
[13] https://20bn.com/datasets/jester
[14] https://gesture.chalearn.org/2016-looking-at-people-cvpr-challenge/isogd-and-congd-datasetsr
[15] https://www.research.ibm.com/dvsgesture/
[16] http://newsscape.library.ucla.edu/

Available implementations and models

Resources for the pipeline components		
Model Name	Available implementation	pre-trained model
Openpose	https://github.com/CMU-Perceptual-Computing-Lab/openpose	Yes
15 Keypoints Is all you Need	NO	NO
Simple Baselines for Human Pose Estimation and Tracking	https://github.com/microsoft/human-pose-estimation.pytorch	yes
DETR	https://github.com/facebookresearch/detr	Yes

Tentative Timeline



During 3 months study and development, It is necessary to catch up with newly coming out methods and study state-of-art in related field, which will help with the the whole working flow. There are generally 5 parts for our project. At the beginning, I will get familiar with Renhen community working flow and play with some useful tools provided by the community, in this part, A Singularity need to be set up. Secondly the tracking system should be implemented for Openpose. Thirdly, The detector should be implemented with a high recall rate and should be tested on the real data. Fourthly, the losing of hand keypoints problem due to the occlusion will be tackled and the gesture detector's precision need also to be improved. Finally, We need to deploy the pipeline and evaluate it in real news video data and a document for the project will be written.

References

[1] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European Conference on Computer Vision*, May 2020.

[2] Okan Köpüklü, Ahmet Gunduz, Neslihan Kose, and Gerhard Rigoll. Real-time hand gesture detection and classification using convolutional neural networks. In *arXiv:1901.10323v3 [cs.CV] 18 Oct 2019.*

[3] Joanna Materzynska, Guillaume Berger, Ingo Bax, and Roland Memisevic. The jester dataset: A large-scale video dataset of human gestures. *ICCV Workshop*, 2019.

[4] Zheng Shou, Dongang Wang, and Shih-Fu Chang. Temporal action localization in untrimmed videos via multi-stage cnns. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[5] Tomas Simon, Hanbyul Joo, Iain Matthews, and Yaser Sheikh. Hand keypoint detection in single images using multiview bootstrapping. *CVPR*, 2017.

[6] Michael Snower, Asim Kadav, Farley Lai, and Hans Peter Graf. 15 keypoints is all you need. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[7] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri. Learning spatiotemporal features with 3d convolutional networks. In *arXiv:1412.0767 [cs], Dec. 2014. arXiv: 1412.0767*, 2014.

[8] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. 2017.

[9] Jun Wan, Yibing Zhao, Shuai Zhou, Isabelle Guyon, Sergio Escalera, and Stan Z. Li. Chalearn looking at people rgb-d isolated and continuous datasets for gesture recognition. *CVPR workshop*, 2016.

[10] Bin Xiao, Haiping Wu, and Yichen Wei. Simple baselines for human pose estimation and tracking. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[11] Y. Zhang, C. Cao, J. Cheng, and H. Lu. Egogesture: A new dataset and benchmark for egocentric hand gesture recognition. *IEEE Transactions on Multimedia (T-MM), Vol. 20, No. 5, pp. 1038-1050, 2018.*

[12] G. Zhu, L. Zhang, P. Shen, and J. Song. Multimodal gesture recognition using 3-d convolution and convolutional lstm. *IEEE Access*, 5:4517–4524, 2017.

Yunfei Zhao

Github: https://github.com/YunfeiZHAO
Website: blog.yunfeizhao.com
Skype ID: live:moizhaoyunfei
Email: yunfei.zhao@etu.utc.fr

EDUCATION

9.2020 - today **Master of COMPLEX SYSTEM ENGINEERING** at UTC
Focus on Machine Learning and Optimization of complex systems
Expected graduation in August of 2021

2.2016 - 9.2020 **Engineering's degree of Data Mining** at UTC
Focus on ML, DL, Optimization, Robotic vision
Computer science and Applied mathematics
Expected graduation in August of 2021

WORK EXPERIENCE

2.2021 - today **Research Internship** at French National Center for Scientific Research

- Road scene analysis by deep learning algorithms
- Combination of sensors for segmentation and object detection
- Data-set construction

2.2019 - 8.2019 **Backend Developer Internship** at Societe Generale

- Java application for monitoring servers states
- CI/CD to automate team development cycle
- Development of a state machine JSON format checker

SKILLS AND QUALIFICATIONS

Programming Languages

Advanced skills	Python, Java, Pytorch, Linux
Basic skills	C++, SQL, Tensorflo, ROS

Languages

Native	Chinese
Advanced	English
Advanced	French

PROJECTS

September 2020 **Master degree projects** Link to project website: https://github.com/YunfeiZHAO/master

- AOS1: Advanced machine learning, posterior distribution inference, etc
- ARS4: Data fusion, Extended Kalman filter, Particular filter, CI filter

September 2020 **Fingerprint recognition** Link to project website: https://github.com/YunfeiZHAO/transfer-lea

- Image pre-processing by center of pixel intensity cropping
- Transfer learning on Restnet50 pre-trained on ImageNet
- Date augmentation

April 2020 **Face detection** Link to project website: https://github.com/YunfeiZHAO/SY32

- Pre-processing of images using Hog, SIFT, etc
- Face detection using sliding window and image pyramids
- Image classification by SVM and tuning Resnet, VGG16 and image detection by Dlib

Feburary 2020 **Learn to fall** Link to project website: https://github.com/YunfeiZHAO/RF-Learn-to-fall

- Reinforcement learning algorithm implementation, DDPG, Actor-Critic
- Opensim, OpenAI model application, Reward function design

September 2019 **Kaggle** Link to project website: https://github.com/YunfeiZHAO/sy09

- Kaggle competition, Biomechanical features of orthopedic patients
- Data cleaning, Dimension reduction, PCA, NCA
- Classical classification algorithms, KNN, Bayes, LDA, GDA, Logistic Regression, Decision Tree random forest, SVM.