National University of Singapore

Phase 1 Report

Thursday 30th April, 2021

*CONTENTS CONTENTS*

Contents

1 Sponsor Company Introduction 3 2 Business Problem Background 4

3 Project Scope 5 3.1 Objectives & Deliverables . . . . . . . . . . . . . . . . . . . . . . . . . . 5 3.1.1 Benchmark Dataset . . . . . . . . . . . . . . . . . . . . . . . . . . 5 3.1.2 Pedestrian Attribute Recognition System . . . . . . . . . . . . . . 6 3.2 Success Measurements . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6

4 Literature Review 8 4.1 Dataset . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8 4.2 Evaluation Criteria . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 10

4.2.1 Label-Based Evaluation . . . . . . . . . . . . . . . . . . . . . . . 10 4.2.2 Example-Based Evaluation . . . . . . . . . . . . . . . . . . . . . . 11 4.3 Model . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 11 4.3.1 Global Image Based Model . . . . . . . . . . . . . . . . . . . . . . 11 4.3.1.1 ACN . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 11 4.3.1.2 DeepSAR and DeepMAR . . . . . . . . . . . . . . . . . 12 4.3.1.3 MTCNN . . . . . . . . . . . . . . . . . . . . . . . . . . . 13 4.3.2 Fixed Region Based Model . . . . . . . . . . . . . . . . . . . . . . 13 4.3.3 Part-Based Model . . . . . . . . . . . . . . . . . . . . . . . . . . . 14

5 Benchmark Dataset 15 5.1 Data Sources . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 15 5.2 Data Structure . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 15

5.2.1 Labels . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 15 5.2.2 Images . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 15 5.3 Attributes . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 16 5.4 Data Labelling . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 18

6 Proposed System Design 20 6.1 System Architecture . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 20 6.1.1 Deployment Architecture . . . . . . . . . . . . . . . . . . . . . . . 20 6.1.2 Offline Training Architecture . . . . . . . . . . . . . . . . . . . . . 21

1

*CONTENTS CONTENTS*

6.1.3 Prototype Architecture . . . . . . . . . . . . . . . . . . . . . . . . 22 6.2 Pre-Processing . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 23 6.3 Feature Extraction & Model Building . . . . . . . . . . . . . . . . . . . . 23

6.3.1 Global Image Based Model . . . . . . . . . . . . . . . . . . . . . . 23 6.3.2 Fixed Region Based Model . . . . . . . . . . . . . . . . . . . . . . 24 6.3.3 Part Based Model . . . . . . . . . . . . . . . . . . . . . . . . . . . 24 6.3.4 Model Evaluation . . . . . . . . . . . . . . . . . . . . . . . . . . . 24 6.4 User Interface Design . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 24

7 Preliminary Result 26

8 Resource Requirements 28 8.1 Cloud Computing . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 28 8.2 On Premise Computing . . . . . . . . . . . . . . . . . . . . . . . . . . . . 29 8.3 Software . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 29

9 Project Timeline 30

10 Potential Challenges 32 10.1 Computation Resource Constraint . . . . . . . . . . . . . . . . . . . . . . 32 10.2 Technical Challenges . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 32 10.3 Data Challenges . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 32 10.4 Other Challenges . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 33

11 References 34 2

*CHAPTER 1. SPONSOR COMPANY INTRODUCTION*

1 Sponsor Company Introduction

NCS Pte. Ltd. (NCS) is a System Integration company that mainly serves the govern ment sector (Ministry of Defence (MINDEF), Land Transport Authority (LTA), Min istry of Home Affairs (MHA), Immigration & Checkpoints Authority (ICA), etc). They have been involved in numerous projects throughout the years ranging from software (e.g. LTA’s MyTransport app, onemotoring.com.sg website) to hardware (e.g. Electronic Road Pricing (ERP) 2.0, ICA passport clearance updates). One area that NCS has been investing in is public security and surveillance, especially in the form of video surveil lance. Recently, NCS has increased its investment in providing outdoor public pedestrian surveillance service due to the increasing demand in the market.

3

*CHAPTER 2. BUSINESS PROBLEM BACKGROUND*

2 Business Problem Background

The usage of video surveillance provides Singapore with the possibility of added security in terms of monitoring/discovering threats to it’s public safety. In an article from Business Insider [1], Singapore is the 11th most surveilled city in the world and 3rd most surveilled city outside of China.

However, in order to detect and track specific pedestrians, there is still a requirement for human investigation on the video data. Furthermore, merely having the surveillance capabilities is insufficient if searchable/trackable insights cannot be extracted from the data. In this case the research area of Pedestrian Attribute Recognition (PAR) is useful to help deal with this extraction of insights. Pedestrian Attributes are “humanly searchable semantic descriptions and can be used as soft-biometrics in visual surveillance, with applications in person re-identification, face verification, and human identification.” [2]

As such, NCS has sponsored us with this project to create a benchmark dataset for outdoor public surveillance and to develop a system to detect pedestrians’ attributes by analysing various outdoor public surveillance data. This will be done using computer vision and other deep learning tools.

4

*CHAPTER 3. PROJECT SCOPE*

3 Project Scope

3.1 Objectives & Deliverables

In this project, the team will be working with NCS with the following two objectives: 1. To create a benchmark dataset

2. To develop a Pedestrian Attribute Recognition system

3.1.1 Benchmark Dataset

The benchmark dataset will be created based on publicly available data. There will be 15 attributes required by NCS. To minimize imbalanced data in the dataset, each attribute will be provided with sufficient labels. The optimal target will be 1000 images per attribute value.

The benchmark dataset will be created and serves as the training dataset to build the classification model. The benchmark dataset can be released for further development such as person re-identification, human identification and attributes mining within the public surveillance research area.

A good benchmark dataset is a cornerstone of model training. With a good benchmark dataset, researchers can significantly reduce the time spent on data collection and labelling required before training a model. For example, “Modified National Institute of Standards and Technology database” (MNIST) is one of the most popular deep learning datasets for handwritten digits recognition [3]. As a “hello world” of machine learning, MNIST has been widely used by data scientists to train and test new architectures or frameworks [3].

Such datasets have many benefits: they can be used to compare existing and new models, It saves other researchers time on the laborious task of data collection. It also demon strates the difficulties of various tasks in one research field.

In the PAR research area, an image can be classified with multiple attributes. With more attributes provided, a greater amount of effort can be put on model training/enhance ment. With a limited amount of certain attributes (e.g. ‘green hair colour’) and attribute types (e.g. ‘carrying item colour’), some PAR attributes cannot be used for data training.

5

*3.2. SUCCESS MEASUREMENTS CHAPTER 3. PROJECT SCOPE*

Hence, in order to build a better recognition system, a more comprehensive benchmark dataset is required.

3.1.2 Pedestrian Attribute Recognition System

The Pedestrian Attribute Recognition system will process the videos/images provided by NCS (from video Surveillance cameras) with bounding boxes, and generate a list of pedestrian attributes. Lastly, it should allow the storing of these attributes and together with the original images, for further uses such as indexing, clustering and searching purposes.

Our PAR system will be integrated and serve as a key component for NCS’s surveillance system. It is designed to include functions such as video splitting, image extraction and attributes identification. The system will be capable of automatically detecting pedestrians by creating bounding boxes and classifying recognized attributes. Having such a system will reduce the manual task(s) required to constantly monitor pedestrians from surveillance data, which is subjected to human factors and has a high risk of oversight or misjudgement.

As a multi-label classification model, the PAR system extracts multiple attributes of each pedestrian. Having the ability to recognize pedestrian attributes like hair colour, hair length, clothing type, carrying items is useful as cues to identify pedestrians distinctly. With a database of multiple features for each pedestrian, pedestrians with specific dressing characteristics can be quickly searched. For example, security personnel or police officers are able to quickly retrieve a person’s image/ security videos based on the description from a witness.

The system can also be deployed with other computer vision systems (like human ac tivity classification, event recognition, person re-identification) to obtain more detailed attributes of the target people in surveillance cameras.

3.2 Success Measurements

The benchmark dataset includes images from public surveillance datasets as well as google images for certain attributes which are found to be lacking in the public datasets, eg. green hair. The team targets to label 1000 images per attribute value for all the 117 requested attribute values.

There are 2 metrics which the team has deemed as important for evaluation, accuracy and speed of responses. In this project the various models developed will be evaluated

6

*3.2. SUCCESS MEASUREMENTS CHAPTER 3. PROJECT SCOPE*

based on model accuracy. To successfully develop the PAR system, the accuracy of the model should be higher than 85%. Also in terms of the speed in which images are scored, the system should be able to provide the list of attributes per image within 10 seconds from the time the image has been uploaded to the system.

7

*CHAPTER 4. LITERATURE REVIEW*

4 Literature Review

4.1 Dataset

The team has reviewed the below 13 datasets mentioned in the survey paper [4], and considered their re-usability for this capstone project.

These datasets have been pre-annotated with various attributes, with 2 major categories:

• Low level attributes, such as hair style and colour, hat, glass etc., which are very specific to certain regions of the pedestrian.

• High level attributes (abstract concepts), such as gender, orientation and age, which do not correspond to certain regions.

Out of the 13, the team has identified the below 3 datasets which could be leveraged on, due to their rich attributes and large image instances:

• PETA (PEdesTrian Attribute dataset) - Constructed from 10 publicly available small-scale datasets which used to research person re-identification.

• RAP & RAP-2.0 (Richly Annotated Pedestrian dataset) - Collected from real indoor surveillance scenarios and 26 cameras are selected to acquire images, Three environmental and contextual factors, i.e., viewpoints, occlusion styles and body parts, are explicitly annotated.

• PA-100K (Pedestrian Attribute dataset) - The PA-100K dataset is constructed by images captured from 598 real outdoor surveillance cameras, is to-date the largest dataset for pedestrian attribute recognition.

The details of the 13 datasets are summarized in Table 4.1.

8

*4.1. DATASET CHAPTER 4. LITERATURE REVIEW*

Table 4.1: Summary of datasets review

| Dataset | # Pedestrians | # Attributes | Finding |
| --- | --- | --- | --- |
| PETA [2] | 19000 | 61 binary, 4  multi-class | Covers most of the re  quired attributes (*∼*90%), PETA’s label setup would be used to minimize the manual data labelling pro cess. |
| RAP [5] | 41585 | 69 binary, 3  multi-class | Covers 80% of the required attributes, use as a poten tial extension of training dataset if required. |
| RAP-2.0 [6] | 84928 | 69 binary, 3  multi-class | Covers 80% of the required attributes, use as a poten tial extension of training dataset if required. |
| PA-100K [7] | 100000 | 26 binary | Covers 20% required at tributes, but this dataset could be used to expand  PETA training data set, (if certain attributes do not have enough instance.) |
| WIDER [8] | 13789 | 14 binary | Covers 50% required at tributes, use as a poten tial extension of training dataset if required. |
| Market-1501 [9] | 32668 | 26 binary, 1  multi-class | Cover more than 70% re quired attributes, use as a potential extension of training dataset if required. |
| DukeMTMC [9] | 34183 | 23 binary | Covers 40% required at tributes, use as a poten tial extension of training dataset if required. |
| PARSE-27K  [10], [11] | 27000 | 8 binary, 2  multi-class | Covers 10% of required attributes |

9

*4.2. EVALUATION CRITERIA CHAPTER 4. LITERATURE REVIEW*

| APiS [12] | 3661 | 11 binary, 2  multi-class | Dataset too small |
| --- | --- | --- | --- |
| HAT [13] | 9344 | 27 binary | Dataset too small and  cover 30% required at  tributes |
| CRP [14] | 27454 | 1 binary, 13  multi-class | Covers 10% of required attributes |
| CAD [15] | 1856 | 23 binary, 3  multi-class | Dataset too small |
| BAP [16] | 8035 | 9 binary | data set too small and at tributes covers only 20% required attributes |

4.2 Evaluation Criteria

There are mainly two evaluation types, which are call as label-based and example-based evaluation criterion by Li, Zhang, Chen, *et al.* [5].

4.2.1 Label-Based Evaluation

Label-based evaluation calculates accuracy and ROC\_AUC for each attribute. It assumes that attributes are independent, which ignores the inter-attribute correlation).

Below shows two main evaluation criteria for label-base evaluation and the formula for mean accuracy.

• AUC\_ROC (based on recall rate and false positive rate) - evaluate the performance of each attribute classification

• mA (mean accuracy) - average accuracy for each attribute

*mA* =12*N*X*L i*=1

*T Pi*

*Pi*+*T Ni Ni*

(4.1)

where *L* is the number of attributes. *T Pi* and *T Ni* are the number of correctly predicted positive and negative examples respectively, *Pi* and *Ni* are the number of positive and negative examples respectively.

10

*4.3. MODEL CHAPTER 4. LITERATURE REVIEW* 4.2.2 Example-Based Evaluation

Example-based evaluation captures better the consistence of prediction on a given pedes trian image [17]

The example-based evaluation criteria which widely used include four metrics: accuracy, precision, recall rate and F1 value, as defined below:

*Accexam* =1*N*X*N i*=1

Precexam =12*N*X*N i*=1

*Recexam* =12*N*X*N i*=1

*|Yi ∩ f* (*xi*)*|*

*|Yi ∪ f* (*xi*)*|*(4.2)

*|Yi ∩ f* (*xi*)*|*

*|f* (*xi*)*|*(4.3)

*|Yi ∩ f* (*xi*)*|*

*|Yi|*(4.4)

*F*1 =2 *·* Precexam *·* Recexam

Precexam + Recexam(4.5)

where *N* is the number of examples, *Yi*is the ground truth positive labels of the *ith* example, *f*(*xi*) returns the predicted positive labels for the *ith* example.

4.3 Model

4.3.1 Global Image Based Model

The following global-image based models are reviewed in this section: • attributes convolutional net model (ACN) [10]

• deep learning based single attribute recognition model (DeepSAR) [18] • deep learning based multiple attribute recognition model (DeepMAR) [18] • multi-task CNN model (MTCNN) [19]

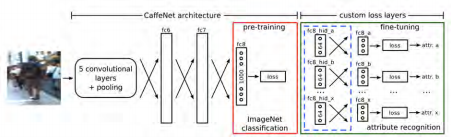
4.3.1.1 ACN

As shown in Figure 4.1, ACN includes a two-stage training. A pre-trained AlexNet on ImageNet has been used as basic feature extraction. The original loss layers are replaced

11

*4.3. MODEL CHAPTER 4. LITERATURE REVIEW*

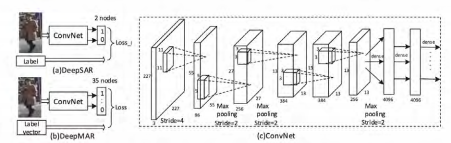
with additional fully-connected (fc) layers. Then a layer with one loss per each attribute is connected after the fc layer.

Figure 4.1: The illlustration of ACN [10]

4.3.1.2 DeepSAR and DeepMAR

DeepSAR and DeepMAR are two CNN based attributes recognition methods. They are created to handle two issues in traditional methods: (a) handcrafted features like color histograms, local binary patterns cannot cope well with video surveillance scenarios; (b) the relationship among pedestrian attributes is ignored.

To solve these problems, Li, Chen, and Huang adopted AlexNet as the backbone network with a changed dense layer. DeepSAR model is proposed to recognize each attribute one by one while DeepMAR takes images and attribute label vectors to consider all the attributes jointly.

Figure 4.2: The basic structure of DeepSAR and DeepMAR

The basic structure of DeepSAR and DeepMAR has been shown in Figure 4.2. The ConvNet in Figure 4.2(c) is a shared network structure between DeepSAR and DeepMAR.

To handle unbalanced data in DeepMAR, the following formula is proposed as an im 12

*4.3. MODEL CHAPTER 4. LITERATURE REVIEW* proved loss function

Loss = *−*1*N*X*N i*=1

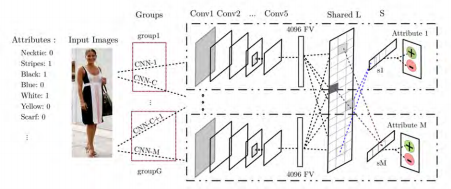
X*L l*=1

*wl*(*yil* log( ˆ*pil*) + (1 *− yil*) log(1 *− p*ˆ*il*)) (4.6)

*wl* = exp *−pl/σ*2 (4.7)

where *wl*is the loss weight for the *lth* attribute. *pl*is the positive ratio of *lth* attribute in the training set. *σ* is a hyper parameter.

4.3.1.3 MTCNN

Figure 4.3: The pipeline of MTCNN

As shown in Figure 4.3, MTCNN is a CNN model with a joint multi-task learning algo rithm. Each CNN will learn a binary nameable attribute. The visual knowledge is shared in a fc layer after each CNN. The shared layer L together with the S layer (CNN-specific weight matrix) form a weight matrix of the last fc layer.

4.3.2 Fixed Region Based Model

Joo, Wang, and Zhu [20] split the original image up into grids and use colour histogram s/Histogram of Oriented Gradient (HoG) [21] as features of the various image patches. On the extracted features, K-means clustering is performed and a part based detector is learned. Lastly an attribute classification is learned for each part (head, upper body, lower body). The use of HoG and colour histograms is inspired by Lazebnik, Schmid, and

13

*4.3. MODEL CHAPTER 4. LITERATURE REVIEW*

Ponce [22]. It was found that having separate classifiers for various parts did improve the attribute classification accuracy on their tested datasets.

4.3.3 Part-Based Model

Other than Fixed Region Based Model to jointly utilize local and global information, Part-Based Model can be implemented to detect objects and generate classification labels based on detected regions.

Part-Based Deep convolutional neural networks have had extensive success in the domain of object detection. Namely Fast RCNN, Mask RCNN and YOLACT 3++ are widely used in Object Detection as well as Instance Segmentation. Part-Based deep convolu tional neural network consists of two key components: a region proposal network (RPN) and a classification sub-network. The RPN works as a sliding window detector by de termining the objects across a set of predefined anchor boxes at each spatial location of the image [23]. After the object proposals are generated, the second stage classifier determines the precise class that each object belongs to. For models Mask RCNN and YOLACT 3++, mask of the object at pixel level can also be generated at the second stage [24].

14

*CHAPTER 5. BENCHMARK DATASET*

5 Benchmark Dataset

5.1 Data Sources

There are 13 pedestrian attribute datasets that are available for use. Of these 13 datasets, some are directly provided, whilst some are provided by researchers after submission of a database license agreement.

An ideal dataset would contain a cropped pedestrian in each image (i.e. where the ROI of the pedestrian constitutes the whole image).

After preliminary statistics and image exploration, the team found that the following datasets meet our needs: PETA, Markter-1501, PA-100K, DukeMTMC & APiS. As sum marized in Table 4.1, PETA covers most of the required attributes (around 90%), PETA’s label setup would be used to minimize the manual data labelling process and effort.

5.2 Data Structure

5.2.1 Labels

All pedestrians are labelled uniquely with the various attributes. Each pedestrian will only be labelled once with attributes even if there are multiple images related to the pedestrian.

5.2.2 Images

Most images are sampled from surveillance video frames. Each image consists of 3 colour channels: Red, Green and Blue. Each pixel of the channel is represented by an 8-bits integer ranging from 0 to 255.

Images are further cropped based into bounding box(es), one for each pedestrian, which can be provided by predefined object detection algorithms (e.g. YOLO). The resolution for each image may vary once it is cropped.

An example of a ROI is shown in Figure 5.1.

15

*5.3. ATTRIBUTES CHAPTER 5. BENCHMARK DATASET *

Figure 5.1: ROI image example

5.3 Attributes

Based on NCS’s requirement, 15 attributes are desired in Pedestrian Attribute Recogni tion. The attributes are mentioned in Table 5.1.

Table 5.1: Summary of attributes

| No | Attribute | Value | Type |
| --- | --- | --- | --- |
| 1 | Gender | personalFemale, personalMale | option |
| 2 | Type of Head  dress | accessoryHat, accessoryFaceMask, accessory Headphone, accessorySunglasses, accessoryHair Band, accessoryKerchief, accessoryMuffler, acces soryNothing | checkbox |
| 3 | Hair Length | hairBald, hairLong, hairShort | option |
| 4 | Hair Colour | hairBlack, hairBlue, hairBrown, hairGreen, hair Grey, hairOrange, hairPink, hairPurple, hairRed, hairWhite, hairYellow | checkbox |
| 5 | Top Type | upperBodyCasual, upperBodyFormal, up perBodyJacket, upperBodyLogo, upper BodyLongSleeve, upperBodyNoSleeve, up perBodyShortSleeve, upperBodySuit, up perBodySweater, upperBodyTshirt, upper BodyOther, upperBodyVNeck | checkbox |

16

*5.3. ATTRIBUTES CHAPTER 5. BENCHMARK DATASET*

| 6 | Top Colour | upperbodyBlack, upperbodyBlue, upperbody Brown, upperbodyGreen, upperbodyGrey, upper bodyOrange, upperbodyPink, upperbodyPurple, upperbodyRed, upperbodyWhite, upperbodyYel low | checkbox |
| --- | --- | --- | --- |
| 7 | Top Clothing  pattern | upperBodyPlaid, upperBodyThickStripes, upper BodyThinStripes | option |
| 8 | Bottom Type | lowerBodyCapri, lowerBodyCasual, lower BodyFormal, lowerBodyHotPants, lowerBody Jeans, lowerBodyLongSkirt, lowerBodyShorts, lowerBodyShortSkirt, lowerBodySuits, lower BodyTrousers | checkbox |
| 9 | Bottom Cloth ing Pattern | lowerBodyPlaid, upperBodyThickStripes, lower BodyThinStripes | option |
| 10 | Bottom Colour | lowerbodyBlack, lowerbodyBlue, lowerbody Brown, lowerbodyGreen, lowerbodyGrey, lower bodyOrange, lowerbodyPink, lowerbodyPurple, lowerbodyRed, lowerbodyWhite, lowerbodyYel low | checkbox |
| 11 | Footwear Type | footwearBoots, footwearLeather  Shoes, footwearSandals, footwearShoes, footwearSneaker, footwearStocking | option |
| 12 | Footwear Colour | footwearBlack, footwearBlue, footwearBrown, footwearGreen, footwearGrey, footwearOrange, footwearPink, footwearPurple, footwearRed, footwearWhite, footwearYellow | checkbox |
| 13 | Type of Carry ing | carryingBackpack, carryingLuggageCase, car ryingMessengerBag, carryingPlasticBags, car ryingSuitcase, carryingNothing, carryingBaby  Buggy, carryingOther, carryingShoppingTro, car ryingUmbrella, carryingFolder | checkbox |
| 14 | Colour of Car rying | carryingBlack, carryingBlue, carryingBrown, car ryingGreen, carryingGrey, carryingOrange, carry ingPink, carryingPurple, carryingRed, carrying White, carryingYellow | checkbox |
| 15 | Others | personalLess15, personalLess30, personalLess45, personalLess60 | option |

17

*5.4. DATA LABELLING CHAPTER 5. BENCHMARK DATASET* 5.4 Data Labelling

After extracting the Region of Interests (ROIs) for individual pedestrian images, the team has applied data a labelling tool Labelbox to label the additional required attributes for each ROI.The team has also combined them with the original PETA annotation to form a more complete list of labels.

The labelling process is completed in the following steps:

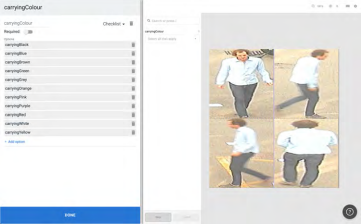
1. The ROI images of the same pedestrian will be collated into one image. This allows the viewing of attributes from different angles and to ensure the accuracy of labelling.

2. Attribute values are configured in Labelbox and must be configured as a checkbox or option depending on whether they are mutually exclusive or not.

3. The labelling work is distributed among the team to improve labelling speed.

4. Once each team member has completed labelling, all labelling data will be exported from Labelbox. The exported data will be further processed for model training (for example into a .mat file).

Figure 5.2 shows a mock-up set-up for attribute CarryingColor and its corresponding attribute values.

Figure 5.2: Labelbox set-up for ’CarryingColor’

The team has already used the above process to label carryingColours for the PETA baseline dataset. It is expected that the team will utilize other related pedestrian public dataset for further model training.

18

*5.4. DATA LABELLING CHAPTER 5. BENCHMARK DATASET*

In order to improve efficiency and save labelling effort, a baseline model DeepMAR\_ResNet 50 trained on PETA can be used to predict labels for other datasets. In this case whilst there is no need to manually label all images, a further step will be done to humanely verify the output labels and correct the mistakes.

19

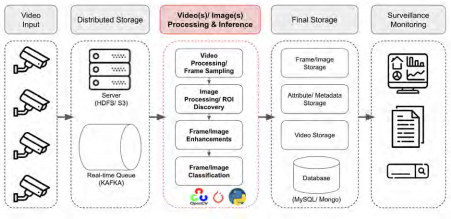
*CHAPTER 6. PROPOSED SYSTEM DESIGN*

6 Proposed System Design

6.1 System Architecture

6.1.1 Deployment Architecture

Figure 6.1 shows the expected deployment architecture, a high level process flow of the larger system that NCS will deploy, which utilizes the system the team will develop. In the section “Processing, Inference & Storage”, the icons OpenCV, PyTorch & Python are used.

Figure 6.1: The illustration of deployment architecture

1. Video Input: Surveillance cameras from the various locations in Singapore will serve as video input into the system

2. Distributed Storage: Various distributed locations will allow for more efficient collection and storage of video inputs. There will be a batch and real-time queue (as required).

3. Processing, Inference & Storage: The Video & Image Processing will be com pleted as per the optimized output from the research. The ROIs will be classified using the saved weights from the trained Neural Network Models & stored for sub sequent searching. (This is the main section that will be developed for NCS.)

20

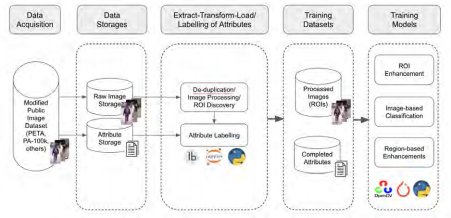
*6.1. SYSTEM ARCHITECTURE CHAPTER 6. PROPOSED SYSTEM DESIGN*

4. Final Storage: The output of the model together with relevant details will be stored in a centralized location for retrieval

5. Surveillance Monitoring: There will be monitoring monitoring with visualiza tion to calculate statistics, create reports and allow for searching of the data. Also, real-time alerts will be pumped into the monitoring system to flag suspicious cases.

6.1.2 Offline Training Architecture

Figure 6.2 shows the offline training architecture, a high level process flow for the gath ering of public datasets, labelling & annotation of attributes, storage & preparation of final training datasets and subsequently modelling & training to obtain the best Pedes trian Attribute Recognition model. In the section “Extract-Transform-Load/Labelling of Attributes” the icons Labelbox, Jupyter Notebook & Python are used. In the section “Training Models”, the icons OpenCV, PyTorch & Python are used.

Figure 6.2: The illustration of offline training architecture

1. Data Acquisition: Data will be acquired from various public datasets (e.g. PETA, PA-100K, etc) Both video and image datasets will be collected to demonstrate the frame sampling process as well as the final output ROIs.

2. Data Storages: This is where initial raw video and images/attributes will be stored pending processing.

3. Extract-Transform-Load/Labelling of Attributes: For videos, this is the phase where the video frames will be sampled into the various images and stored. For images, this is where the images will be processed (enhanced/sharpened etc.)

21

*6.1. SYSTEM ARCHITECTURE CHAPTER 6. PROPOSED SYSTEM DESIGN*

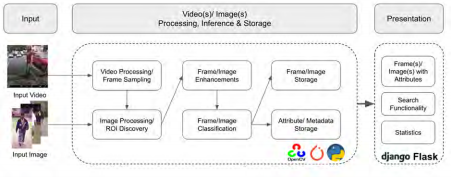
and de-duplicated (select/combined) for easy labelling. Also the full image will be split into smaller ROIs for PAR.

4. Training Datasets: This is where the final processed dataset is stored

5. Training Models: In this phase, the features will be created and used in the model for classification. Subsequently evaluation will be done on the test set.

6.1.3 Prototype Architecture

Figure 6.3 shows the prototype architecture, a high level process flow for the demo envi ronment to showcase the UI and system developed by the team. In the section “Processing, Inference & Storage”, the icons OpenCV, PyTorch & Python are used.

Figure 6.3: The illustration of prototype architecture

1. Input: The demo environment will allow for input of images/ video as well as provide sample images/ videos.

2. Processing, Inference & Storage: The Video & Image Processing will be com pleted as per the optimized output from the research. The ROIs will be classified using the saved weights from the trained Neural Network Models & stored for sub sequent searching.

3. Presentation: Sample frames from the video/ ROIs will be shown with the Pedes trian Attributes.

22

*6.2. PRE-PROCESSING CHAPTER 6. PROPOSED SYSTEM DESIGN* 6.2 Pre-Processing

In the pre-processing, video frames must be sampled and the Region of Interest (ROI) extracted for labelling by the model, in this step image enhancement can also be done to improve performance. Below are some example methods that can be used to fulfill the pre-processing step.

To extract frames, one proposed method is to use the OpenCV function: cv2.VideoCapture(), to read the frames in for analysis. To sample the images, it is possible to choose every Nth frame to be labelled, or a random number of frames can be selected, other more intelligent methods to sample might also be devised.

To extract the ROI, one proposed method is to use a pre-trained human detection algo rithm to extract the bounding box. The Haar Cascade provided within OpenCV itself can serve as a starting point for extraction of the ROIs.

As the obtained ROIs can sometimes be quite pixelated or blurry, methods could be used to sharpen and refine the image for better classification downstream. One proposed method is to do interpolation (e.g. 2D nearest-neighbour, Bilinear, Bicubic) between the various pixels to generate a larger image size from a small one. Another would be sharpening the edges and salient features using kernels or to do histogram equalization to reduce contrast between pixels in an image.

6.3 Feature Extraction & Model Building

Traditionally, several model types have been proposed for the modelling experiments on Pedestrian Attribute Recognition, namely, Support Vector Machine, Convolutional Neu ral Network, and other proven multi-streams neural network models. For this Capstone project, the team has decided to use deep neural networks based on the following three model types: global image based, fixed region based and part region based. The final model might be a combination (via embedding/voting) of the following types or a single type which has the best evaluation performance.

6.3.1 Global Image Based Model

In a Global Image Based Model the entire ROI is used to create features and the whole image is labelled with the various attributes. Global images represent the whole area of the ROI generated. From our research, the DeepMAR\_ResNet-50 model uses ResNet-50 as a feature extractor. Feature maps are generated based on global images.

23

*6.4. USER INTERFACE DESIGN CHAPTER 6. PROPOSED SYSTEM DESIGN* 6.3.2 Fixed Region Based Model

Fix region based model focuses on handling geometric variation which require manual part annotation. The model cuts the global ROIs into different subregions based on the predefined size W \* H, and additional annotations for subregions are given by rules. For example, in the global image, subregion size of 50 \* 50 from coordinations (0, 0) is the area of the head. Then feature extractors are implemented in subregions for related classifiers. Certain subregions can only be labelled with certain attributes (e.g. the head region cannot be labelled with “lowerBodyShorts”)

6.3.3 Part Based Model

Similar to fix region based, part based focuses on partitioning of the global ROI as well. However, in part based model, model uses openCV or other available predefined models such as OpenPose or Mask RCNN are used to extract the mask of the object. Following which, feature extractors are implemented in the mask regions to classify labels.

6.3.4 Model Evaluation

The evaluation of the model will be tested on two tiers. Firstly, the team will compute the label-based accuracy to evaluate if attribute options are given correctly. Then it will compute the example-based accuracy to evaluate if the attributes are given correctly to individual pedestrians.

To select the final deployed model, there is a tradeoff between accuracy and computation time. On one side, there is a need to keep the high accuracy of the model as the labelled results will be used for further development. On the other side, the model should predict the classes within the acceptable range of time.

6.4 User Interface Design

Whilst this project serves as a component in NCS surveillance system to identify the key attributes of pedestrians, the team also imagines possible UI designs which can be adopted by NCS. In the UI design there should be monitoring and searching features (for technical operators).

For the demo purpose, the team will provide a sample UI which allows users to up load images/videos. Once the images/videos are uploaded, the system will process them

24

*6.4. USER INTERFACE DESIGN CHAPTER 6. PROPOSED SYSTEM DESIGN*

sequentially and display the respective pedestrian image with the attributes sorted by confidence (highest to lowest). Attribute values with a confidence score below a certain threshold will be highlighted.

The images and the extracted attributes are stored in the backend database in pairs, the user will be able to search the images by certain attribute values, and the system should find the respective list of pedestrians.

25

*CHAPTER 7. PRELIMINARY RESULT*

7 Preliminary Result

The team has managed to train the DeepMAR\_ResNet-50 model on the PETA dataset with 100 epochs.

Evaluation result on test set:

• Label-based evaluation:

– mA: 0.8418

• Example-based evaluation:

– Acc: 0.7912

– Prec: 0.8779

– Rec: 0.8546

– F1: 0.8661

After that, the team scored 1 sample image with the trained model weights. The results are shown in Figure 7.1 with details in Table 7.1 on the 35 sample attribute values. The attribute values with classification score higher than 0 have been recognized by the model.



Figure 7.1: Demo Results

26

*CHAPTER 7. PRELIMINARY RESULT*

Table 7.1: Full result of the demo

| Attribute Values | Classification Score | Possibility |
| --- | --- | --- |
| personalMale | 16.361866 | 100.00% |
| footwearShoes | 16.15862 | 100.00% |
| personalLess45 | 14.028587 | 100.00% |
| carryingOther | 12.2588005 | 100.00% |
| accessorySunglasses | 10.547851 | 100.00% |
| lowerBodyFormal | 9.439131 | 99.99% |
| upperBodyFormal | 8.817605 | 99.99% |
| upperBodyTshirt | 8.585138 | 99.98% |
| upperBodyShortSleeve | 8.039649 | 99.97% |
| lowerBodyShorts | -7.581402 | 0.05% |
| carryingMessengerBag | -9.136396 | 0.01% |
| upperBodyVNeck | -9.26455 | 0.01% |
| lowerBodyShortSkirt | -9.876684 | 0.01% |
| lowerBodyCasual | -10.03392 | 0.00% |
| upperBodyPlaid | -10.335828 | 0.00% |
| lowerBodyJeans | -10.583626 | 0.00% |
| personalLess60 | -10.6857395 | 0.00% |
| accessoryNothing | -11.014259 | 0.00% |
| upperBodyCasual | -11.27231 | 0.00% |
| upperBodyLogo | -11.36648 | 0.00% |
| footwearLeatherShoes | -11.542577 | 0.00% |
| accessoryMuffler | -12.048079 | 0.00% |
| carryingNothing | -12.194274 | 0.00% |
| lowerBodyTrousers | -12.490903 | 0.00% |
| personalLess30 | -12.528797 | 0.00% |
| personalLarger60 | -13.317648 | 0.00% |
| upperBodyThinStripes | -13.4465475 | 0.00% |
| accessoryHat | -13.784597 | 0.00% |
| upperBodyOther | -14.380968 | 0.00% |
| carryingPlasticBags | -15.072072 | 0.00% |
| hairLong | -15.264586 | 0.00% |
| footwearSandals | -15.355138 | 0.00% |
| carryingBackpack | -15.947015 | 0.00% |
| footwearSneaker | -16.52043 | 0.00% |
| upperBodyJacket | -17.84243 | 0.00% |

27

*CHAPTER 8. RESOURCE REQUIREMENTS*

8 Resource Requirements

8.1 Cloud Computing

As an image classification model, computing resources can largely affect the training speed. To balance the training speed and cost, a system with a powerful GPU is required.

High-Performance Computing (HPC) provided by NUS IT will be used to train and test the model.

Here is the technical specification:

• CPU: INTEL Xeon X5650 2.66GHz Hexacore

• GPU: NVIDIA Tesla M2090 6GB GDDR5 512 cores

• RAM: 48GB

The team estimates the cost based on Google Cloud Platform with the following setup [25]:

• 100 total hours per month

• VM class: regular

• Instance type: n1-standard-8

• Region: Iowa

• GPU dies: 1 NVIDIA TESLA V100

• GPU’s Cost: SGD 355.57

• GCE Instance Cost: SGD 54.48

• Total available local SSD space 1x375 GB

• Estimated Component Cost: SGD 415.94 per 1 month

• Total Estimated Cost: SGD 415.94 per 1 month

It will cost SGD 2500 for 6 months (May-October).

28

*8.2. ON PREMISE COMPUTING CHAPTER 8. RESOURCE REQUIREMENTS* 8.2 On Premise Computing

It can also run on a personal desktop with a large GPU RAM. Here is a customized Alienware desktop price for reference only [26].

• CPU: 9th Gen Intel® Core™ i9 9900K (8-Core, 16MB Cache, Overclocked up to 4.7GHz across all cores)

• GPU: NVIDIA® GeForce® RTX 2080 SUPER™ 8GB GDDR6 (OC Ready)

• Hard Drive: 512GB M.2 PCIe NVMe SSD (Boot) + 2TB 7200RPM SATA 6Gb/s (Storage)

• Memory: 16GB Dual Channel HyperX™ FURY DDR4 XMP at 2666MHz • Total Price: SGD 4498

8.3 Software

The software used in this project is summarised in Table 8.1.

Table 8.1: Summary of software

| Software | Description |
| --- | --- |
| Labelbox | Labelbox is an end-to-end platform to create and man age high-quality training data all in one place. |
| Python | Python is an interpreted, high-level, general-purpose programming language. The packages like OpenCV and PyTorch are used during the model building |

29

e

n

i

l

e

m

Ti

t

c

e

j

o

r

P

9

.

e

n

i

l

e

tim

t

c

e

j pro

the

s

w

sho

9.2

e

r

u

Fig

d

an

9.1

e

r

u

Fig

g

n

i

w

o

l

fol

e

h

T

30

1

e

n

o

t

s

e

l

Mi

:

1

.

9

e

r

u

g

i

F

31

2

e

n

o

t

s

e

l

Mi

:

2

.

9

e

r

u

g

i

F

*CHAPTER 10. POTENTIAL CHALLENGES*

10 Potential Challenges

10.1 Computation Resource Constraint

As NCS is unable to provide computing resources, the team has explored using the High Performance Compute (HPC) capability from NUS. However there is some need to famil iarize with the structure and means to run models in the HPC. Also, as the HPC is shared by many users in NUS, the project might be adversely affected by scheduled/unscheduled maintenance and downtime.

10.2 Technical Challenges

It is a requirement of NCS that the PyTorch framework is used to build and train the Pedestrian Attribute Recognition model. This is to fit in with other deep learning frame works and modules used at NCS. However as PyTorch is new to the team, some ramp up might be required to understand the framework and build/train PyTorch based mod els effectively. Also, the level of technical depth available in PyTorch is more than in TensorFlow, whilst this allows for more fine-tuning and tweaking it is supposedly also harder.

Furthermore, there are few publicly known PyTorch based models for the Pedestrian Attribute Recognition research area which can be used as reference architectures for comparison and improvement.

10.3 Data Challenges

1. Attribute Imbalance in public datasets:

It was found that the number of attribute values for the public datasets are unequal, for example, most of the bags colours are black with the other colours (e.g. pink, purple, yellow) being quite rare. With a high degree of imbalance in our attribute values, the model performance might be aversely affected.

2. Significant effort required for data preparation:

There is a need to manually label a large amount of images for each attribute value 32

*10.4. OTHER CHALLENGES CHAPTER 10. POTENTIAL CHALLENGES*

which is missing/lacking from the public datasets (e.g. “green hair colour”). Often it is required to search for images containing these attribute values from other websites / image databases and prepare them for use, which is undoubtedly tedious and onerous.

3. Personal preference/bias for attribute value:

As the pedestrian images are generally of low resolution, it is sometimes very diffi cult to determine the exact attribute value provided. For example, a dark looking bag might be seen as either black or blue depending on the illumination factor. Also, occlusion and items just happening to be in the hands of a person might appear to be carried by the individual.

10.4 Other Challenges

The team also faces other softer challenges such as the need to juggle between full time jobs and the part-time masters. This is especially difficult during periods of strict deadlines and submissions when there are overlapping priorities. Furthermore, as each stays in separate locations in Singapore and because of COVID-19, it has been difficult to meet in person for syncs and discussions.

33

*CHAPTER 11. REFERENCES*

11 References

[1] B. Insider, *Singapore is the 11th most-surveilled city in the world – but it doesn’t even come close to China: Report)*, https://www.businessinsider.sg/singapore-is the - 11th -most - surveilled - city -in - the - world - but -it - doesnt - even - come - close - to - china-report, [Online; accessed 18-April-2020], 2020.

[2] Y. Deng, P. Luo, C. C. Loy, and X. Tang, “Pedestrian attribute recognition at far distance”, in *Proceedings of the 22nd ACM international conference on Multimedia*, 2014, pp. 789–792.

[3] L. Deng, “The mnist database of handwritten digit images for machine learning research [best of the web]”, *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 141– 142, 2012.

[4] X. Wang, S. Zheng, R. Yang, B. Luo, and J. Tang, “Pedestrian attribute recognition: A survey”, *arXiv preprint arXiv:1901.07474*, 2019.

[5] D. Li, Z. Zhang, X. Chen, H. Ling, and K. Huang, “A richly annotated dataset for pedestrian attribute recognition”, *arXiv preprint arXiv:1603.07054*, 2016. [6] D. Li, Z. Zhang, X. Chen, and K. Huang, “A richly annotated pedestrian dataset for person retrieval in real surveillance scenarios”, *IEEE transactions on image processing*, vol. 28, no. 4, pp. 1575–1590, 2018.

[7] X. Liu, H. Zhao, M. Tian, L. Sheng, J. Shao, S. Yi, J. Yan, and X. Wang, “Hydraplus net: Attentive deep features for pedestrian analysis”, in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 350–359.

[8] Y. Li, C. Huang, C. C. Loy, and X. Tang, “Human attribute recognition by deep hierarchical contexts”, in *European Conference on Computer Vision*, Springer, 2016, pp. 684–700.

[9] Y. Lin, L. Zheng, Z. Zheng, Y. Wu, Z. Hu, C. Yan, and Y. Yang, “Improving person re-identification by attribute and identity learning”, *Pattern Recognition*, vol. 95, pp. 151–161, 2019.

[10] P. Sudowe, H. Spitzer, and B. Leibe, “Person attribute recognition with a jointly trained holistic cnn model”, in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2015, pp. 87–95.

[11] P. Sudowe and B. Leibe, “Patchit: Self-supervised network weight initialization for fine-grained recognition.”, in *BMVC*, vol. 1, 2016, pp. 24–25.

34

*CHAPTER 11. REFERENCES*

[12] J. Zhu, S. Liao, Z. Lei, D. Yi, and S. Li, “Pedestrian attribute classification in surveillance: Database and evaluation”, in *Proceedings of the IEEE international conference on computer vision workshops*, 2013, pp. 331–338.

[13] G. Sharma and F. Jurie, “Learning discriminative spatial representation for image classification”, 2011.

[14] D. Hall and P. Perona, “Fine-grained classification of pedestrians in video: Bench mark and state of the art”, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5482–5491.

[15] H. Chen, A. Gallagher, and B. Girod, “Describing clothing by semantic attributes”, in *European conference on computer vision*, Springer, 2012, pp. 609–623. [16] L. Bourdev, S. Maji, and J. Malik, “Describing people: A poselet-based approach to attribute classification”, in *2011 International Conference on Computer Vision*, IEEE, 2011, pp. 1543–1550.

[17] M.-L. Zhang and Z.-H. Zhou, “A review on multi-label learning algorithms”, *IEEE transactions on knowledge and data engineering*, vol. 26, no. 8, pp. 1819–1837, 2013. [18] D. Li, X. Chen, and K. Huang, “Multi-attribute learning for pedestrian attribute recognition in surveillance scenarios”, in *2015 3rd IAPR Asian Conference on Pat tern Recognition (ACPR)*, IEEE, 2015, pp. 111–115.

[19] A. H. Abdulnabi, G. Wang, J. Lu, and K. Jia, “Multi-task cnn model for attribute prediction”, *IEEE Transactions on Multimedia*, vol. 17, no. 11, pp. 1949–1959, 2015. [20] J. Joo, S. Wang, and S.-C. Zhu, “Human attribute recognition by rich appearance dictionary”, in *The IEEE International Conference on Computer Vision (ICCV)*, Dec. 2013.

[21] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection”, in *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR’05)*, IEEE, vol. 1, 2005, pp. 886–893.

[22] S. Lazebnik, C. Schmid, and J. Ponce, “Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories”, in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’06)*, vol. 2, 2006, pp. 2169–2178.

[23] G. Brazil, X. Yin, and X. Liu, “Illuminating pedestrians via simultaneous detection & segmentation”, in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 4950–4959.

[24] K. He, G. Gkioxari, P. Dollár, and R. Girshick, “Mask r-cnn”, in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969. [25] Google. (2020). Google cloud platform pricing calculator, [Online]. Available: https: //cloud.google.com/products/calculator/#id=6ffda818-f575-4bc4-8b75-f59dee1de0fe.

35

*CHAPTER 11. REFERENCES*

[26] Dell. (2020). Alienware aurora pro gaming desktop with 9th gen intel cpu | dell singapore, [Online]. Available: https : / / www . dell . com / en - sg / shop / desktop - computers / alienware - aurora - r9 - desktop / spd / alienware - aurora - r9 - desktop / w21905002sgw10cfls?view=configurations&configurationid=98a6e5d1-2482-48ea 82f1-4d3b8b6e82f1.

36