Biometrics: Body Shape Recognition

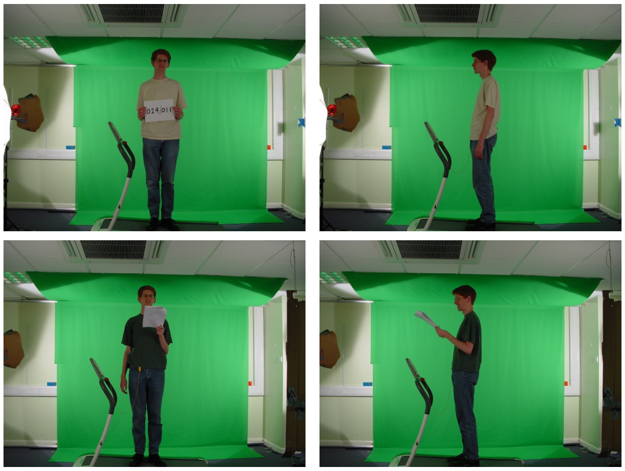
David Jones
School of Electronics and Computer Science
University of Southampton
Southampton, United Kingdom
dsj1n15@ecs.soton.ac.uk

***Abstract*—The role of a biometric system is to identify subjects for purposes including access control and authentication. This report studies the use of body-shape features as a means to classify subjects. The approach taken utilises state-of-the-art neural networks for pose estimation and semantic segmentation before applying more traditional techniques to define feature vectors. The approach has a CCR of 91% and EER of 33%. Increasing performance was largely handicapped by the irregularities in the small dataset and lack of subject repetition.**

***Index Terms*—Identification, Body Shape, Limb, Shape Descriptors, Pose Estimation, Semantic Segmentation**

I. Introduction

A requirement for a biometric system is being able to to differentiate subjects. Existing image based systems typically use facial recognition [1], whereas video systems may aim to categorise people's gait [2]. This report identifies the feasibility of body-shape features on a small image dataset with side-on and front-on views of each subject. A second dataset with some subjects repeated is then then used to assess classification performance. Datasets are subsets of the Large Southampton Gait Database [3] (Figure 1).



(a) Front (b) Side

Fig. 1: Example of subject present in both datasets.

II. Method

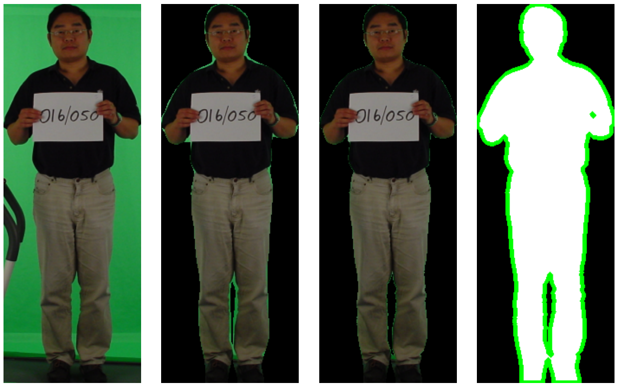
This section details feature-vector generation and classification methods. In addition to those mentioned, the Python implementation utilised the common libraries: NumPy[1], OpenCV[2] , PyTorch[3], and SKLearn[4].

A. Features

The final feature-vectors are made from a selection of measurements and moments; those used depend on the view (front/side). Input images were resized to have a height of 600px whilst keeping the same aspect ratio to reduce computation.

1) Subject Mask: The first stage of feature extraction was to find which pixels of the input belong to the subject; this is referred to as image segmentation. The subject mask was generated in two stages:

1) Semantic segmentation performed by a Convolutional Neural Network (CNN), specifically DeepLabV3 ResNet101 [4]. Torchvision's[5] implementation was used.
2) This did not follow the body contours completely (Figure 2b), so a further green-screen removal stage was done in the HSV colour space. (Figure 2c)



(a) (b) (c) (d)

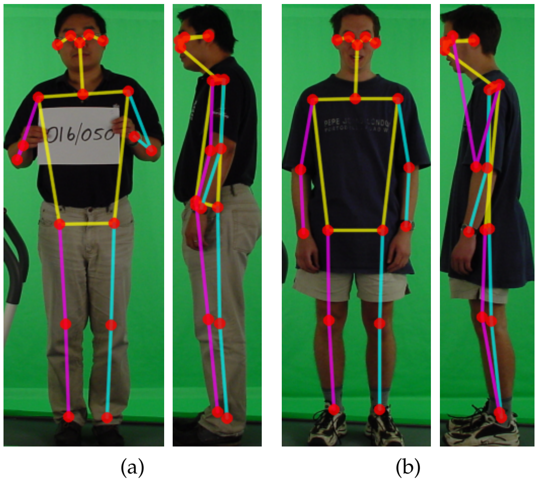
Fig. 2: Masking example: input, semantically segmented, green screen removed, silhouette with extracted contour points (left–right). Note that the example images are cropped, the actual system accepts full images.

[1] Numpy, https://numpy.org/
[2] OpenCV, https://opencv.org/
[3] PyTorch, https://pytorch.org/
[4] scikit learn, https://scikit-learn.org/
[5] Torchvision, https://pytorch.org/docs/stable/torchvision

This method lead to perfect mask generation for all subjects. Traditional methods including active contours typically failed where there was crossover with the treadmill.

Hu Moments were generated from the full-body mask to form part of the feature vector for the side-on views. These are preferable to area/perimeter as they are invariant to translation, scale and rotation. A *height* metric was also calculated by finding the difference between the lower and upper bounds of the mask.

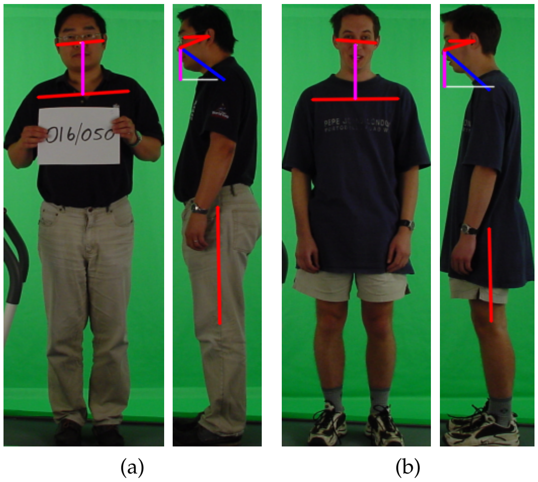
2) Keypoint Estimation: Fitting a virtual skeleton to a body is known as pose estimation. This method identifies body keypoints with understanding of the underlying semantics that dictate their position; this allows extraction where limbs are occluded or overlap. Torchvision's implementation of Keypoint R-CNN [5] was used to extract 17 XY keypoints. These were reordered and an extra keypoint was extrapolated for the chest to match OpenPose's[6] data format (Figure 3).



(a) (b)

Fig. 3: Keypoint estimation on two subjects.

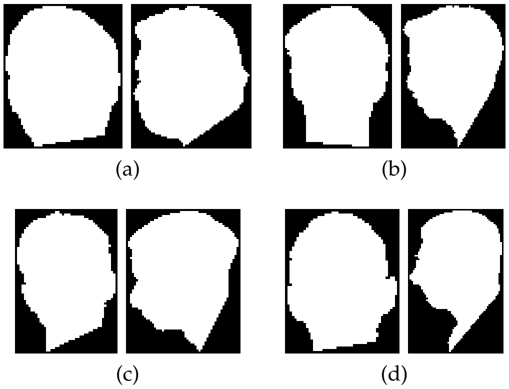
These keypoints were accurate where positioning was not subjective, e.g. for noses, but not always where many positions would be acceptable e.g. the pelvis. Additionally, because Z data is not estimated, distances between points cannot take into account depth. Points were therefore mostly used to orientate other features. However, where distances were relatively consistent it was possible to add them to the feature vector; those used are labelled in Figure 4. It would be preferable if these measurements were scale invariant in case the subject stood a different distance from the camera, unfortunately, normalising off one measurement made the features highly non-differentiable. The *view* (*direction*) was determined using the width between shoulder points and the relative position of the nose.



(a) (b)

Fig. 4: Keypoint measurements for each view. red, pink, blue lines indicate where Euclidain distance, Y Manhatten distance and angle (from the horizontal) are used respectively.

3) Head Mask Extraction: Some subjects have different clothes between the validation and testsets; this reduces effectiveness of the full-body mask. Extracting the head portion provides a more consistent shape.



(a) (b) (c) (d)

Fig. 5: Example head masks (front–side for each subject).

There are no keypoints to make neck extraction straightforward. Simple methods may attempt to find the thinnest point of the mask however this fails with hair and does not take into account the angle of the neck for the side-on view. The following method was used:

1) Extract the contour of the full-body.
2) Trace from the shoulder keypoints to the edge of the contour on the left/right (side-on) and top (front-on) to find two starting points.
3) Plot the X positions following the contour upwards.
4) Use peak/trough detection to find the turning points (where the head begins).
5) Remove proceeding points of the contour to create a contour of just the head.
6) Convert this contour to a mask.

[6] OpenPose, https://git.io/JfVsu

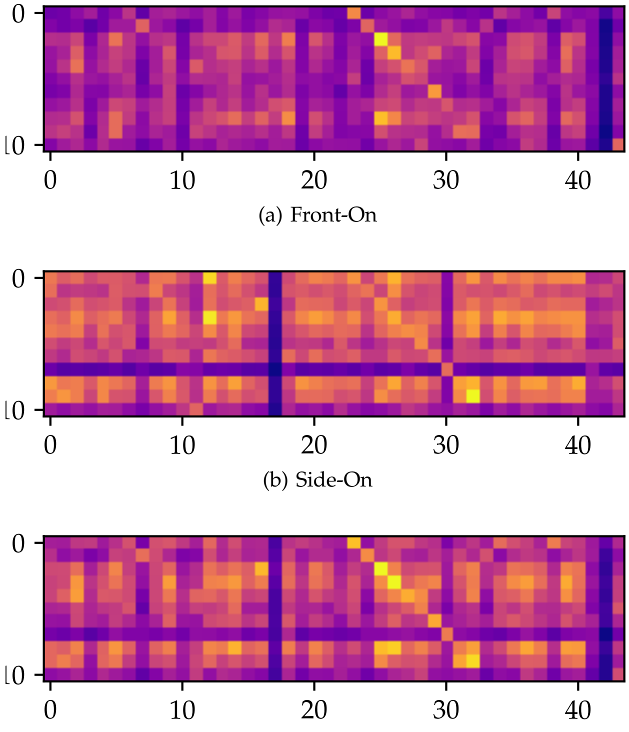
Hu Moments are generated for head masks for all views.

B. Classifier

The classifier expects both front-on and side-on views of a subject where the view is inferred as part of feature extraction. The different views use different feature vectors so cannot be directly compared. To this end separate k-nearest-neighbour (k-nn) classifiers are used for each view – $k = 1$ due to lack of repetition in the dataset. These are trained on the *authorised* user set with feature normalisation applied to keep the subject distances between 0 and ∼1. These are combined by taking the mean of the all distance vectors generated; this allows classification via further views or repetitions of the same view. A threshold calculated via the EER is applied to limit false classifications. SVMs were considered, however as their advantages are curtailed by the small dataset, and because they reduce the relevance of the easily comparable distance metric, k-nn was preferable.

III. Results

To assess the system the `test` dataset (11 classes) were treated as the *authorised* subjects, whereas the `train` dataset (44 classes) were treated as the *input* subjects. The resultant feature vector distances and top matches are shown in Figure 6 and Figure 7 respectively.



(a) Front-On

(b) Side-On

(c) Combined

Fig. 6: Log Euclidean distances between subjects. Each row represents an *authorised* subject against all *inputs*. See Figure 9 for correct classifications. Temperature indicates closeness (Yellow [Close], Purple [Far]).

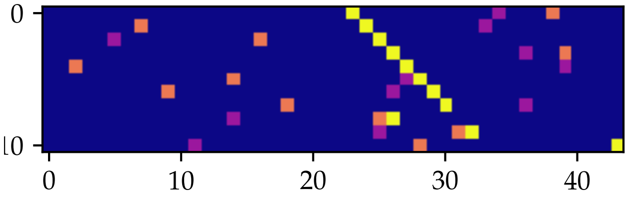


Fig. 7: Top 3 matches (1–3 [Yellow, Orange, Purple]) for each *authorised* class using the combined metric.

A. Accuracy

The side-on and front-on classifiers can both be tuned to achieve 63% CCR individually, however a focus is placed on assessment of the combined system. Metrics were calculated as:

- **CCR:** Correct Recognition Rate – Percentage of *input* subjects who were closest to their corresponding *authorised* subject.
- **T-CCR:** Threshold-CCR – CCR but taking into account threshold calculated via EER.
- **FAR:** False Accept Rate – Percentage of *input* subjects not in the *authorised* dataset who were within the allowed threshold of an *authorised* subject.
- **FRR:** False Reject Rate – Percentage of *input* subjects who were not within the threshold of their corresponding *authorised* subject.

The EER of 33% was calculated from the intersect of the FAR and FRR (Figure 8), yielding a threshold distance of 0.15. Figure 9 shows the accuracy of the system is 91% (10/11), reduced to 64% when thresholding (7/11).

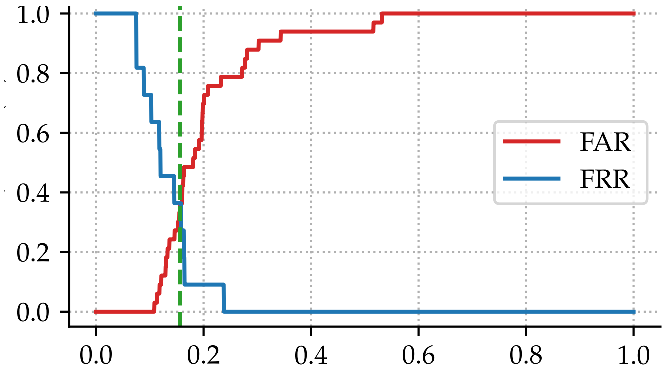


Fig. 8: EER Plot – Intercept = (0.15, 0.33)

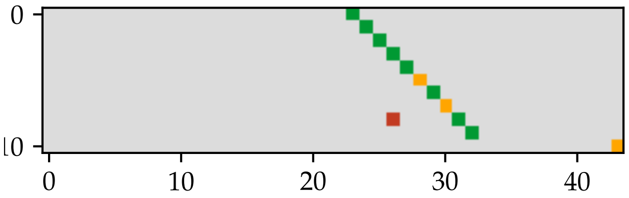


Fig. 9: T-CCR Plot – True positives, false negatives and false positives shown by green, orange, red respectively. Note where classified incorrectly, green identifies the correct class.

B. Speed

A single recognition took 10.2s running on a CPU (Intel i7 8700K). Much of this time was taken up by the segmentation and keypoint detection performed by neural networks. However, these were integrated with GPU support so when tied with a Nvidia GeForce 1080 Ti this time was 0.6s.

IV. Discussion

A. Performance

The results are promising, suggesting that subjects can be differentiated based on their body shape. However, performance is not comparable to facial recognition systems, with modern implementations achieving EERs of 0.08% and older systems still achieving 4.1% [6]. The comparably high EER can largely be attributed to the inter-class variance of the features not being high as shown by Figure 6; this implies low confidence in predictions. The results are mostly influenced by the head Hu Moments and height measurement, with which a CCR of 64% (7/11) could be achieved. The incorporation of other measurements lead to the CCR of 91%, however it could be deemed that these additions lead to overfitting. Likewise, it is common for a single extracted feature to match badly; this increases the overall distance which leads to the EER threshold having to be increased. This is however hard to assess due to lack of different validation / test sets. The often negative effect of the full-body Hu Moments highlight the issues of recognising by body shape for clothed subjects. Avoiding usage of clothed areas is a challenge in itself and reduces the number of usable features. The overall running time suggests that real-time performance is viable for stills but not video, however, it should be noted that the system was not optimised.

B. Improvements

There are many features that could be implemented in the future including but not limited to: edge analysis via Gabor wavelets or training a CNN to find border patterns. More importantly, effort should be placed on making the system fully invariant to subject-to-camera distance and to reduce the effect of limb occlusion/depth. This could be achieved by combining a multi-camera setup that can handle 3D pose estimation with body measurement normalisation (i.e. body ratios).

Classification could also be more adaptable. The side-classifier typically had a higher degree of confidence than the front-classifier. Performance could likely be improved by weighting the components of the combined classifier based on confidence level. This would be especially useful if other views could be added.

Speed-ups can be achieved by implementing a single CNN for both segmentation and pose estimation; reducing computation by half. Furthermore additional caching could be implemented to stop features being generated multiple times for a single feature-vector.

V. Conclusion

A system has been proposed that is able to classify individuals purely on body-shape; this reduces dependency on a subject's face or video source of movement. The system's low-confidence (similarities in distances) highlight the issues with recognising clothed subjects and indicate further views or averaging across a larger dataset would improve performance substantially. Other improvements have been detailed in the discussion.

References

[1] B. O. Omoyiola, "Overview of biometric and facial recognition techniques," vol. 20, pp. 1–5, 07 2018.

[2] M. S. Nixon, T. Tan, and R. Chellappa, *Human Identification Based on Gait*. Springer US, 2006. [Online]. Available: http://dx.doi.org/10.1007/978-0-387-29488-9

[3] J. Shutler, M. Grant, M. S. Nixon, and J. N. Carter, "On a large sequence-based human gait database," in *Fourth International Conference on Recent Advances in Soft Computing (01/11/02)*, 2002, pp. 66–72, event Dates: November 2002. [Online]. Available: https://eprints.soton.ac.uk/257901/

[4] L. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," *CoRR*, vol. abs/1706.05587, 2017. [Online]. Available: http://arxiv.org/abs/1706.05587

[5] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," 2017.

[6] W. Crumpler. (2020, April) How accurate are facial recognition systems – and why does it matter? CSIS. [Online]. Available: https://www.csis.org/blogs/technology-policy-blog/how-accurate-are-facial-recognition-systems-—and-why-does-it-matter (Accessed 24/05/2020).