**SMART BODY MASS INDEX SYSTEM**

**PROJECT REPORT**

**COMP 4983 – CAPSTONE PROJECT**

**EXCELLENCE SOWUNMI**

**100158476**

**PROFESSOR: DR MAHBUB MISHU**

**APRIL 6, 2023**

**INTRODUCTION**

No matter what their occupation, everyone can benefit from a smart BMI system because it can track your height and has additional capabilities that may be used for your health in more complex situations. There are various BMI calculators out there that provide fundamental input and output features. I don't intend to make anything as basic as those, though. In my version, I will use a picture taken through a camera lens to determine the user's height, add their weight, and then compute the outcome. As Furthermore, I would like to include an option to input your age and an algorithm to check if your BMI is right for your age group.

**BACKGROUND**

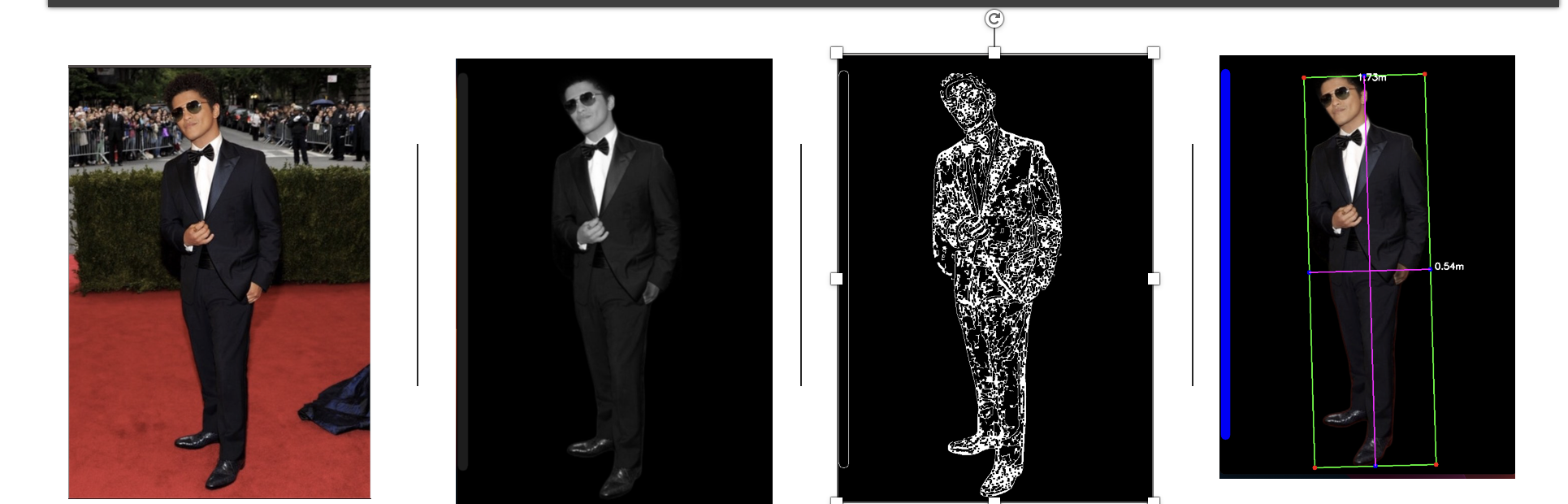
The issue of determining a person's body height from a single uncalibrated photograph is covered in this text ([BenAbdelkader et. al., 2008](https://www.paperdigest.org/paper/?paper_id=doi.org_10.1109_afgr.2008.4813453)). It suggests a human-height estimation technique that derives an estimate from a person's video clip using visual geometry and feature learning. Additionally, it investigates how to rebuild an object's surface using the word structure of light. Finally, using the known physical measurement of an image, a new measuring method provides height measurements and related errors. An image that only depicts the upper body area and an image without a reference length in the background scene are two difficult examples that haven't been covered in the literature before. It might be interesting to investigate a method for estimating a person's body height from a single uncalibrated photo. The work ([BenAbdelkader et. al., 2008](https://www.paperdigest.org/paper/?paper_id=doi.org_10.1109_afgr.2008.4813453)) combines well-known ideas from projective geometry and single-view metrology with an existing probabilistic and statistical understanding of human anthropometry. A novel measuring algorithm derives height measurements and the errors related to those measurements from the known physical measurements of an image. In the article (Liu & Zhao, 2018) the word structure of light is used to recreate the surface of an item. The final stage of the research process focuses on real-time, high-precision, high-speed reconstruction in three dimensions.

**METHODS AND MATERIAL**

****In this project, we propose a human height estimation method that performs a calibration using an infused reference line. The dimensions of our reference line are in a quantifiable unit and are simple to locate in a picture. Additionally, it is distinctively distinguishable as it is in the left corner of the image and as a distinct colour. The width of the line is set to 0.044 meters with a thickness of 20 for visibility, while the length of the line is set from the top of the face of the person to somewhere close to the feet of the person. To determine the size of the human in a picture from this line, the pixels to the metric ratio is the ratio of the human width to the line width.

I use the imutils package to import Python packages, parse command line arguments, load an image from disk, perform edge detection, find contours, initialize the pixels to the metric ratio, examine each contour region, and compute the rotated bounding box of the image. Firstly, I take the background in the image out using Rembg. a Python package that enables the removal of image backgrounds. create the midpoint helper method, which is used to determine the midpoint between two sets of (x, y) coordinates. We then obtain our image from the disk, convert it to grayscale, and then apply a Gaussian filter to smooth it out. To fill in any gaps between edges in the edge map, we then execute edge detection combined with a dilation + erosion. We locate contours that match up with the items on our edge map. Then, these contours are sorted from left to right. (Allowing us to extract our reference object) Additionally, we set the value of the pixels to the metric ratio Examining each of the outlines is the next step. Then we begin iteratively looping through each of the distinct outlines. We reject the contour region if it is too small and compute the rotational bounding box of the image instead. For OpenCV 2.4 and 3.0, we utilize the cv2.cv.BoxPoints function and order the coordinates from top-left to top-right to bottom-left to top-left. Next, we outline the object in green and add small, red circles to represent the vertices of the enclosing box rectangle. The bottom-right midpoint is calculated next, after which we compute a sequence of midpoints between the top-left and top-right locations. To initialize the pixels to the metric ratio variable, we compute the Euclidean distance between our sets of midpoints, draw blue midpoints on our image, then look at our reference object. The height distance will be stored in the dA variable, and the width distance will be stored in dB. To approximate the number of pixels per inch, we determine whether the pixels to the metric ratio variable has been initialized and divide dB by the provided 0.044. To gauge the size of items in an image, this is done. Lastly, we compute the object's dimensions and show the resultant data.

**RESULT AND DISCUSSION**

****

We were able to accurately determine the dimensions of the human in our photograph and record the size on the output image. But due to an improper angle and a lack of a precise 90-degree perspective, the object measurements are not entirely accurate, producing distorted dimensions. The infused reference line is also not an accurate measurement, but it gives a reasonable basis for improvements. There is not always going to be a reference height in an image so this process could help estimate a human height in an image. The removal of the background from the image helped increased the result as it removed any excessive and unnecessary contours/edges.

**CONCLUSION**

We were able to measure the human in our image with accuracy, but the measurements of the objects were off because of an incorrect angle and a lack of a true 90-degree perspective. The removal of the backdrop from the image improved the outcome of the human height estimation using the implanted reference line. In the future, we could make a lot of improvements, like tackling the problem of image Zoom or video files or pictures with more than one human.

**REFERENCES**

BIBLIOGRAPHY

C. BenAbdelkader and Y. Yacoob, "Statistical body height estimation from a single image," 2008 8th IEEE International Conference on Automatic Face & Gesture Recognition, Amsterdam, Netherlands, 2008, pp. 1-7, doi: 10.1109/AFGR.2008.4813453.

S. Dokthurian, W. Rattanapitak and S. Wangsiripitak, "Human Height Estimation Using Visual Geometry and Feature Learning," 2021 IEEE International Symposium on Circuits and Systems (ISCAS), Daegu, Korea, 2021, pp. 1-5, doi: 10.1109/ISCAS51556.2021.9401250.

Lee, D., Kim, J., Jeong, S. C., & Kwon, S. (2020). Human Height Estimation by Color Deep Learning and Depth 3D Conversion. *Applied Sciences*, *10*(16), 5531. MDPI AG. Retrieved from <http://dx.doi.org/10.3390/app10165531>

Liu, W., & Zhao, Y. (2018, November 1). Height Estimation of Target Objects Based on Structured Light. *Advances in Multimedia*, *2018*, 1–9. <https://doi.org/10.1155/2018/4189125>

*About The Haar-Cascade Face Detection Algorithm*. (n.d.). About the Haar-Cascade Face Detection Algorithm | Face Detection With HC. https://viswalahiri.github.io/Face\_Detection\_Haar\_Cascade/scripts/

Gullbadhar, A. (2022, December 12). *Remove background from Images using Python and AI*. Medium. https://levelup.gitconnected.com/remove-background-from-images-using-python-and-ai-149a6985e478

Rosebrock, A. (2016, March 28). *Measuring size of objects in an image with OpenCV - PyImageSearch*. PyImageSearch. https://pyimagesearch.com/2016/03/28/measuring-size-of-objects-in-an-image-with-opencv/