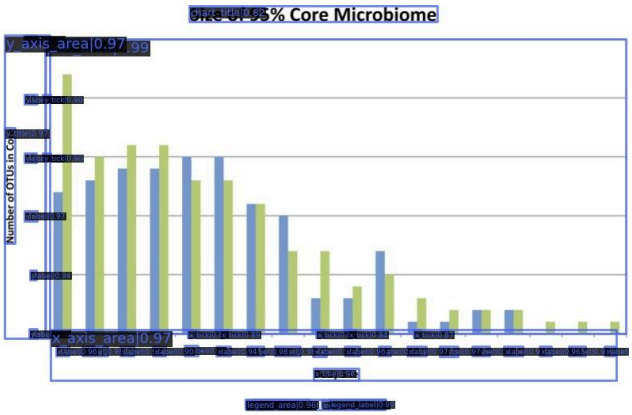
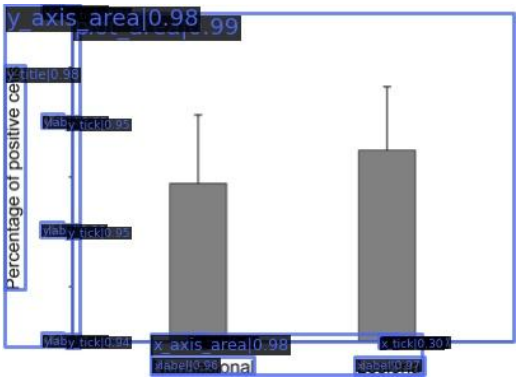
Chang Han (u1472415), Ishrat Jahan Eliza (u1472593)

Methodology

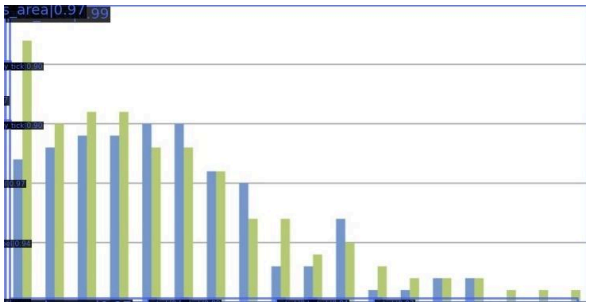
To identify the colored part and data types and to convert the plot images later to generate texture plots, we need to separate the regions of interest. In our case, the region of interest is only the plot area, but there are no other plot elements, such as titles, ticks, labels, legends, etc.

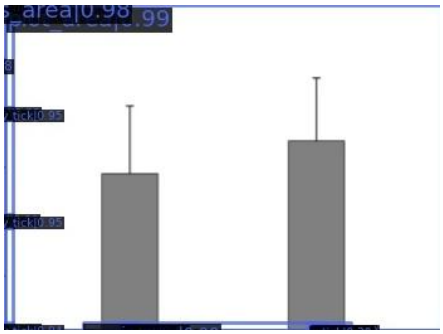




We detect the chart/ plot elements in this project using the CACHED (Context-Aware Chart Element Detection) [1] model. We used the R-CNN framework to integrate a local-global context fusion module consisting of visual context enhancement and positional context encoding from CACHED. The final model is called the cascade R-CNN framework. The model classifies 18 chart elements, such as x_title, y_title, x_ticks, y_ticks, plot_area, xlabel, ylabel, x_axis_area, y_axis_area, tick_grouping, chart_title, etc. and detects them. The attached images are two examples of the final result of chart element detection using the Cascade R-CNN model. The dataset we used here is from the PubMed Central (PMC) Chart Dataset that was released and updated with the Chart Competition in ICDAR 2019 [5] and ICPR 2022 [6].

The Cascade R-CNN model only detects the chart element but doesn't generate separate images for each component from the parent chart. We need the plot area as input for the second part of our project. Thus, we enhanced the method to fulfill the purpose. We used the OpenCV module in Python to read the image that detected areas in bounding boxes. The openCV converts the image into a numpy array and extracts the region of





interest, i.e., the plot area in our case, using the array slicing techniques. Figures at the left show two separate plot areas from two separate input images that we discussed in the first paragraph.

For the second part of our project, we aim to: **(i) identify the different colored parts of visualizations** and (ii) **determine the data types that the colors are encoding**. We began by exploring existing tools. ChartSense [3] uses an interactive approach to extract data from visualizations, while ChartOCR [2] goes further with fully automated data extraction. However, both methods face accuracy issues—ChartOCR has a mean error of around 0.1, and ChartSense requires significant manual effort to correct the data. Additionally, these approaches rely heavily on training corpora and manually set rules, limiting their effectiveness to specific charts. This is not ideal for our application, where we aim to support a wider variety of chart types.

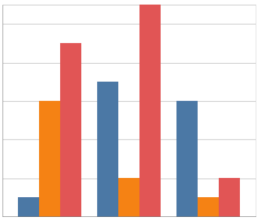
Fortunately, unlike previous methods, we don't need to extract precise data values but rather identify 'data patterns' (quantitative, ordinal, or categorical) and segment the different colored parts. To address this, we employed a simplified approach. We first detected the boundaries of the colored areas using the SLIC superpixel algorithm, recognizing that some charts use continuous color gradients. To handle this, we set a threshold and apply Gaussian blur to classify these charts appropriately.

$$\Delta E_{00} = \sqrt{\left(\frac{\Delta L'}{k_L S_L}\right)^2 + \left(\frac{\Delta C'}{k_C S_C}\right)^2 + \left(\frac{\Delta H'}{k_H S_H}\right)^2 + R_T \frac{\Delta C' \Delta H'}{S_C S_H}}$$

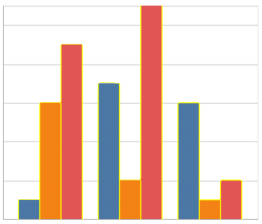
where:

- $\Delta L'$, $\Delta C'$, $\Delta H'$ are the adjusted differences in lightness, chroma, and hue.
- S_L, S_C, S_H are scaling factors.
- R_T is a rotation term that accounts for interactions between chroma and hue differences.

Next, we transformed the colors into the **CIELAB** color space, which aligns more closely with human visual perception. In CIELAB, equal color differences correspond more accurately to perceived differences compared to RGB and HSV. We then used the CIEDE2000 Color Difference Formula [4] (shown on the left) to calculate the differences between adjacent colors. If the difference exceeds a threshold L, we classify it as categorical data; if the difference is smaller, we classify it as quantitative or ordinal data.



original image



segments with fluorescent boundary

mean distance is 20.9 > 10

categorical

Chang Han (u1472415), Ishrat Jahan Eliza (u1472593)

Future Plan

We find the detection boundary, and the boundary labels generated by the cascade R-CNN model generate some overlaps, which results in some generation of the noisy region of interest in our outputs. This noisy region works as a hurdle to detecting individual colored parts of the plot area later. In the future part of the project, we plan to train the model with less noisy bounding boxes to generate cleaner images.
We have generated output based on around 5k images of bar plots. We also want to test the result with other plot types, such as pie charts and scatter plots, to see how it performs for the datatypes, primarily how the second part of the project (detecting colored parts) works with the current implementation. Additionally, we want to explore other popular datasets used in other chart element detection research work.
Our final work is mapping texture to the colored parts of the charts and visualizing the final output.

References

[1] Yan, P., Ahmed, S., Doermann, D. (2023). Context-Aware Chart Element Detection. In: Fink, G.A., Jain, R., Kise, K., Zanibbi, R. (eds) *Document Analysis and Recognition - ICDAR 2023*. Lecture Notes in Computer Science, vol 14187. Springer, Cham. https://doi.org/10.1007/978-3-031-41676-7_13
[2] J. Luo, Z. Li, J. Wang and C. -Y. Lin, "ChartOCR: Data Extraction from Charts Images via a Deep Hybrid Framework," *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, Waikoloa, HI, USA, 2021, pp. 1916-1924, doi: 10.1109/WACV48630.2021.00196.
[3] Daekyoung Jung, Wonjae Kim, Hyunjoo Song, Jeong-in Hwang, Bongshin Lee, Bohyoung Kim, and Jinwook Seo. 2017. ChartSense: Interactive Data Extraction from Chart Images. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17). Association for Computing Machinery, New York, NY, USA, 6706–6717.
[4] Sharma, Gaurav, Wencheng Wu, and Edul N. Dalal. "The CIEDE2000 color‐difference formula: Implementation notes, supplementary test data, and mathematical observations." *Color Research & Application: Endorsed by Inter‐Society Color Council, The Colour Group (Great Britain), Canadian Society for Color, Color Science Association of Japan, Dutch Society for the Study of Color, The Swedish Colour Centre Foundation, Colour Society of Australia, Centre Français de la Couleur* 30.1 (2005): 21-30.
[5] Davila, K., Tensmeyer, C., Shekhar, S., Singh, H., Setlur, S., Govindaraju, V.: *ICPR 2020 - competition on harvesting raw tables from infographics*. In: Del Bimbo, A., et al. (eds.) ICPR 2021. LNCS, vol. 12668, pp. 361–380. Springer, Cham (2021). doi: 10.1007/978-3-030-68793-9_27.
[6] Davila, K., Xu, F., Ahmed, S., Mendoza, D.A., Setlur, S., Govindaraju, V.: *ICPR 2022: challenge on harvesting raw tables from infographics (chart-infographics)*. In: 2022 26th International Conference on Pattern Recognition (ICPR), pp. 4995–5001. IEEE (2022)