David Owen

- Associate Professor of Computer Science, *Messiah College*, Grantham, PA.
- This presentation: https://dvon.github.io/home/aha/aha.pdf

Disclaimer

- I'm not a historian...
    - And *computer vision* is not my research area.
    - This is a subject I'm interested in, motivated to learn so that I might be able to teach a course in the future.
- *I'd be glad to learn from you*, if anyone in the audience has experience in this area.

Background

- Census data available (from, e.g., Ancestry.com).
    - But some information missing...transcribed by volunteers.
    - Would be *beneficial to have an alternative* way of converting to images to text.
- Specific goal is to *speed up process of converting scans.*
    - From Harrisburg census forms, from about 100 years ago.
- *Optical character recognition* (OCR) software works well...
    - For machine-generated text.
    - For handwritten text, if training data is available.
- But what about *handwritten text, without training data*?
    - If content is relatively simple, limited to a small number of possibilities?

Background (2)

- OCR software is designed to recognize characters in image data.
    - ...to *"read" the text in a picture.*
    - It's particularly hard to recognize *handwritten* characters.
- But maybe we don't need to recognize characters.
    - If we had *a way to group similar images.*

- *A human user* could interpret a single image, representative of a group.
    - The software could then apply that interpretation to all of the images in the group.
- Work up to this point makes use of OpenCV and scikit-image software libraries for computer vision and image processing, via Python (3) bindings.

Overview

- *Generate a template* from a composite of scanned census forms.
- *Create cell images.*
    - Choose a form, choose a column.
    - Crop images, based on template.
- *Process images*, prepare for comparison.
    - Delete boundaries, based on template.
    - Reconnect broken lines.
    - Weight towards darkest central region.
- For prepared images from same column...
    - *Divide into similarity-based groups.*
    - *Enable user to visualize results*, verify groupings, assign value to group, etc.

Scans

- What do *the scans we have* look like?
    - Example scan...
- How would we automatically *generate a template*?
    - To specify where to crop cell images, where within cell image are boundaries to be deleted.
    - Composite image, template image...
- Will *a single template* will be sufficient?
    - Is there enough consistency between scans?
    - Bad scan...

Cells

- *Create an image for each cell.*
    - Choose a form, choose a column.
    - For each cell in column, copy and crop individual cell image.
    - Ownership columns...
- Prepare cell image for comparison.
    - *Delete* (horizontal and vertical) *cell boundaries*, based on template image.
    - One attempt at deleting boundaries, another attempt...
    - Reconnect broken lines. (*Working on an algorithm* for this; not sure how successful or necessary it will be.)

Similarity Groups

- Considering *all cell images from a column...*
    - Across multiple form images, eventually?
    - Ownership columns (again)...
- *Divide into similarity-based groups.*
    - Using *scikit-image comparison functions*?
    - K-Means approach, used in recognition of, e.g., Chinese characters, has also been suggested.

User Interaction

- Create annotated version of form image.
    - *Mark images* to show which group they belong to.
    - Provide indication of confidence level for similarity-based groups...*How similar are cells within group?* How distinct are cells in different groups? Which cell is a good representative?
- Human user interaction...
    - Verify similarity, based on sample of less-similar images from within a group.
    - *Assign textual value*, based on representative image, for all images in group.