

Post-Capture Processing

Page Segmentation Correction (layout reclassification and reflow)

Originally Decapod was designed to provide a UI to allow for page segmentation correction. However, this is going to change because:

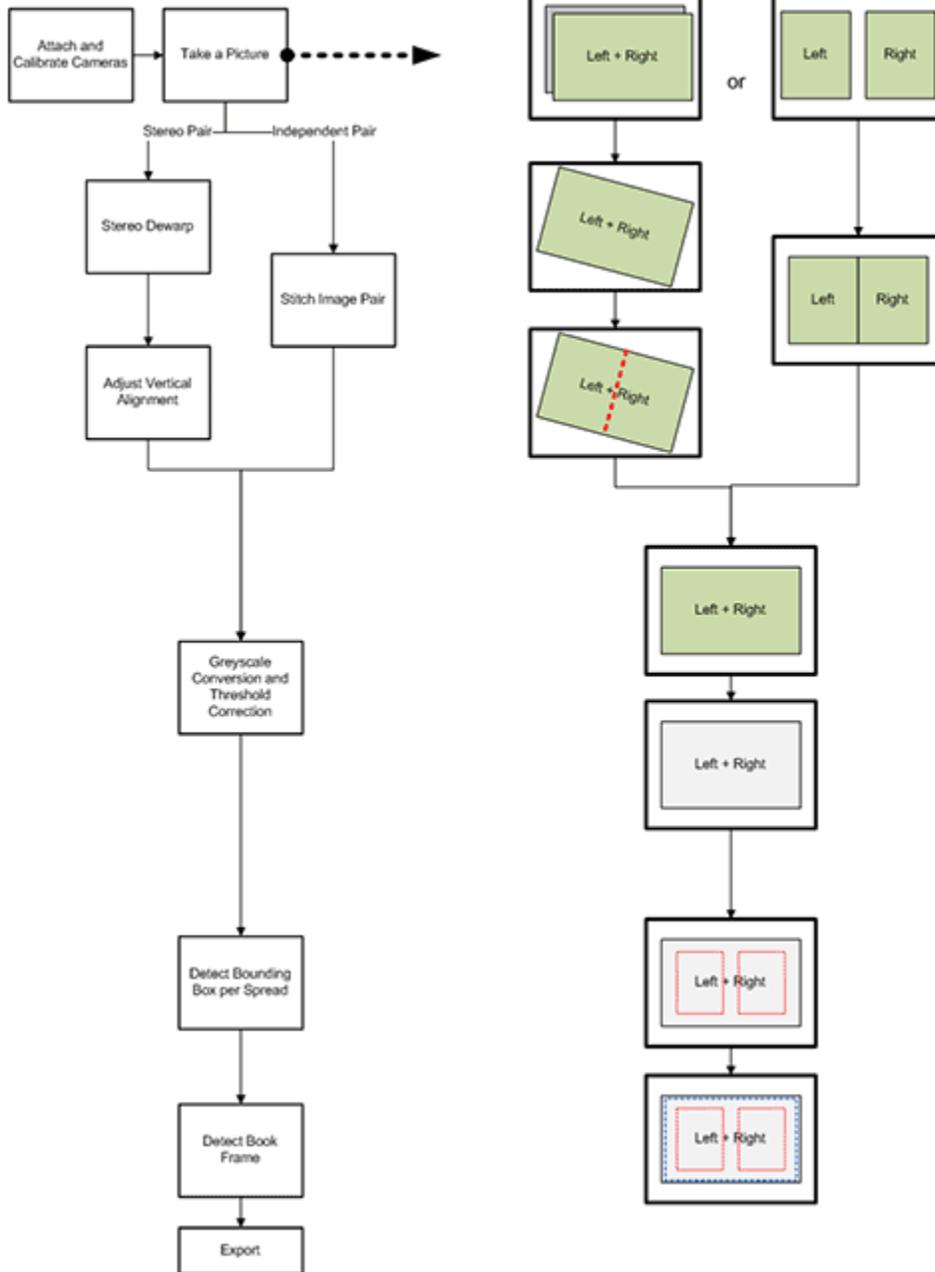
- output results not very good if the input images are poor. Therefore more important to get input images as good as possible, rather than correcting segmentation first.
- focus on improving input images to get better results (better page seg., font generation, and ocr)
- Page Segmentation Correction is something that may be covered in the next Decapod related project.
- From a UI design and development perspective, Page Segmentation Correction is going to be very challenging and consume a lot of resources with respect to the pixelwise colour maps used in page segmentation.

Getting Good Results

In order to get the best results from Ocropus, the following criteria should be met (or as close as possible):

1. angle of rotation per page
2. height and width of content per page
3. height and width of the book frame (a bounding box that encompasses all pages in #2).
4. image threshold adjusted (black and white balancing)

Decapod Functional Workflow



Axis correction

Axis correction helps improve results of OCR and font generation as it helps ensure that characters are in the proper orientation to be processed.

- by drawing a line marking the spine of a page, we gain two very important bits of information: axis orientation, and book frame height. (this gives us #1 and half of #3 in the list of criteria).
- there is some margin for error (1/2 to 1 degree)

Implementation 1: User driven correction

1. Stereo dewarped image is presented to the user. A grid is overlaid the image so user can see how straight the axis is.
2. User draws a line down the spine fold to correct rotation.
3. Image is automatically rotated.
4. User can now: redraw the line, revert to original, or accept the rotation.
5. There is a checkbox next to the accept button which allows the user to apply the rotation to subsequent images.

Implementation 2: Automatic correction

1. Image is stereo dewarped.
2. Image is then deskewed automatically.

GIMP has an auto deskew function which may simplify things. However, deskew may already be in decaPod-genpdf?

- See: <http://www.cubewano.org/gimp-deskew-plugin/>

Implementation 3: Automatic with user correction

1. Image is stereo dewarped.
2. Image is then deskewed automatically.
3. Stereo dewarped-deskewed image is presented to the user. A grid is overlaid the image so user can see how straight the axis is.
4. User draws a line down the spine fold.
5. Image is automatically rotated.
6. User can now: redraw the line, revert to original dewarped-deskewed image (the same as in step 3), or accept the rotation.
7. There is an option that allows the user to apply the rotation to subsequent images.

Content Bounding Box

A content bounding box is a way of specifying what is important and what is not. This effectively is like a crop. The box helps improve OCR results by removing any information that is outside the main content area which may create erroneous results.

Note: Axis correction should be performed before this step as drawing a box over content that is not square with the axis is difficult.

Implementation 1: User specified boxes

1. An axis corrected image is presented to the user. Two identically sized boxes are already drawn on the left and right side of the image.
2. User can reposition the left and right boxes independently by dragging the box.
3. User can resize the boxes by grabbing an edge or corner. ? Design decision: are changes mirrored on the opposite box as well, or is resizing independent?
4. User can reset the boxes to its original size and position (as in Step 1), accept the boxes, or cancel.
5. There is an option to allow the user to apply the boxes' positions and sizes to all subsequent pages.

Implementation 2: Automatic

Use an automatic cropping algorithm to crop irrelevant information. Will need to test this solution to see how tight the box will be.

Specifying the book frame bounding box

The book frame bounding box helps specify the final dimension of the output file. Since each spread there may be different sized content bounding boxes, the book frame bounding box is large enough in both width and height to contain them all.

In a way the book frame bounding box can be thought of as:

- width = max width (spread 1 to spread end)
- height = max height (spread 1 to spread end)

This can be automatically generated without user interaction.

Threshold correction

Threshold correction improves the contrast of black and white text so that they are effectively processed.

Implementation 1: Client-Side Threshold Slider

1. Image of the page spread is presented to the user.
2. A slider / spinner is provided so they can increase or decrease the threshold.
3. User can now: revert to original image, or accept the threshold value.
4. There is also an option to apply the threshold correction to all subsequent images.

Implementation 2: Server-Side Threshold Chooser

1. Image of the page spread is presented to the user in the center of the interface.
2. Around the original image are 8 threshold alternatives.
3. User chooses one of the alternatives to accept.

Implementation 3: Automatic

Possible options:

- [ImageMagick - Triangle, fuzzy, and more...](#)
 - See here for a comparison of the different threshold methods: http://www.fmwconcepts.com/imagemagick/threshold_comparison/index.php
- GIMP's auto threshold

GIMP has an auto threshold function which may be callable. Also, existing ocropus / decapod scripts may already do this? (decapod-stitching does some threshold correction already?).

Implementation 4: Automatic with Client or Server Side

- Perform automatic threshold and then present a client-side or server side threshold adjustment for fine-tuning.

Notes:

- rotation correction is likely the first step always because it's hard to draw boxes on an image that rotated horribly left or right.
- axis correction, content bounding box, and page frame bounding box can be streamlined by offering users the ability to apply changes to just the current spread or to all following spreads.

Interesting Scenario

- the above corrections and bounding boxes assume that all images come from a single book captured in a single setting.
- What if the user imports a series of photos into the book which have larger content and book frame boxes than the rest of the images already captured?
- does resizing of input content to fit existing content too much for this project? This is already done well by other image editing software.

On this Page

- [Page Segmentation Correction \(layout reclassification and reflow\)](#)
- [Getting Good Results](#)
- [Axis correction](#)
- [Content Bounding Box](#)
- [Specifying the book frame bounding box](#)
- [Threshold correction](#)
- [Interesting Scenario](#)

Implementation Research and Testing

- [Automatic Content Detection Research and Testing](#)
- [Image Rotation Research and Testing](#)
- [Image Threshold Research and Testing](#)