









MOCHA - Minimal Optical Coffee Height Analysis

Machine Vision

Java

Machine Vision

Digital Image Capture Cost v's Quality

- Used relatively cheap digital camera
- Image manipulation using filters and operators Median, threshold, max, edge detection, skeletonizing, ...

Inference

Pulling knowledge from a digital image Coffee volume Hough transform Classic detects straight lines General detects precalculated arbitrary shapes



Image Filters - Threshold





Image Filters - Skeletonize





Coffee Volume - How it should have been



MOCHA 6



Coffee Volume - How it is



MOCHA 7

Hough Transform



Classic Hough Transform

Detects straight lines



Classic Hough Transform

► Find edges



Explanation of Classic Hough Transform

- Each point in an image votes for all the lines that go through it y = mx + c
 - $x\cos(\theta) + y\sin(\theta) = r$





Classic Hough Transform

Hough space



Explanation of Classic Hough Transform



General Hough Transform

Detects arbitrary shapes

Will add ability to detect rotated and scaled shapes





Shapes must be built up from an image Must have a reference point Stores shape as an array of x, y difference pairs Each point in a digital image votes to say that it may

be part of a shape



Shape of Coffee Pot





General Hough Space



Explanation of General Hough Transform

Voting array is thresholded

- Any points remaining above the threshold limit indicate the likely presence of the shape in the original image
 - They actually represent reference points
 - These points can be used to map the shape back into image space

Shape Mapped Back on to Image



Conclusions

- Image manipulation toolkit in Java
- Machine vision can be done on the cheap
- Shape building functions
- Working classic and general Hough transform implementations

Need to play with to determine parameters for successful useage