Automatic Text Recognition for Imagery in the Wild

ACC program proposal #562

Peter Cho (Group 106) & Davis King (Group 104)

December 2012

LINCOLN LABORATORY
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

Technical Challenge: Digital Imagery Exploitation

- **Digital pictures are being collected at rates far exceeding human exploitation capabilities**
  - **Billions of photos now exist in online archives like Flickr**
  - **72 hours of video are currently uploaded to YouTube every minute**
- **Algorithms are needed to flag pictures of special interest for human analysis**
- **Automatic text recognition would provide valuable metadata & context for otherwise unstructured input imagery**

Q: What is the setting of this picture?

Q: What language is spoken by locals in this picture?

Q: In what town was this picture shot?

Information Inferable from Imagery Text

- **Topic domains (e.g. from business names & advertisements)**
  - **Spatial: Indoor vs outdoor, urban vs rural**
  - **Temporal: Winter vs summer, daytime vs nighttime**
  - **Settings: Shops, libraries, crowds**
- **Cultural contexts (e.g. from alphabet recognition)**
  - **Language identification**
  - **Nationality determination**
- **Approximate to precise camera geolocations (e.g. from road signs)**
  - **Street address detection**
  - **Landmark name geofingerprinting**

Q: What is the setting of this picture?

A: Hardware store interior

Q: What language is spoken by locals in this picture?

A: Chinese

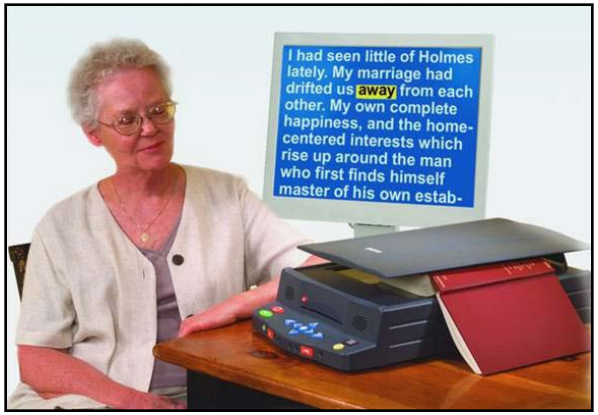
Q: In what town was this picture shot?

A: New Hanover, NC

Text Recognition Difficulty vs Image Gathering Cooperation

Data processing complexity

Document character recognition



e.g. Book scanning

Augmented reality

e.g. iPhone translation app

"Imagery in the wild"

e.g. Random internet photos

A priori camera uncertainty

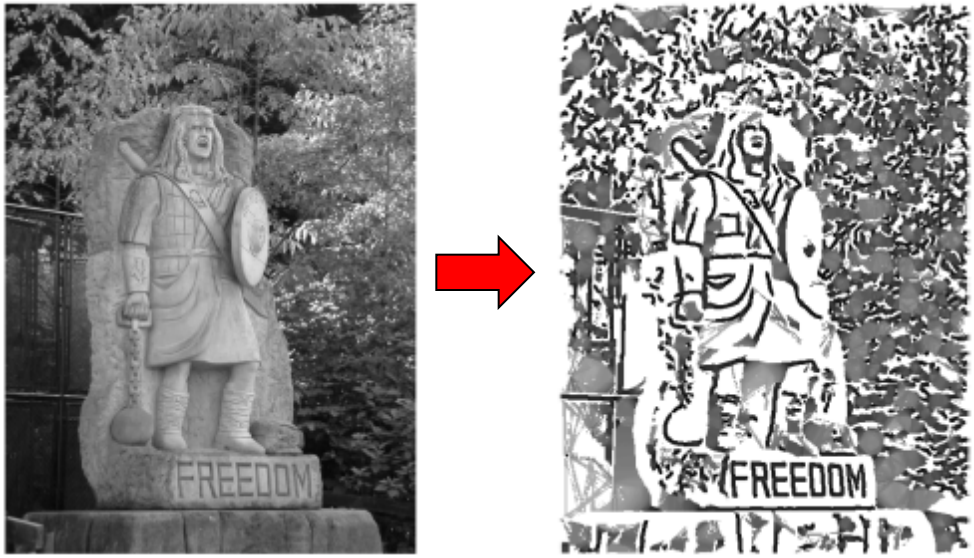
Outline

- **Prior art**
- **Program plan**
- **Schedule, budget & follow-on potential**

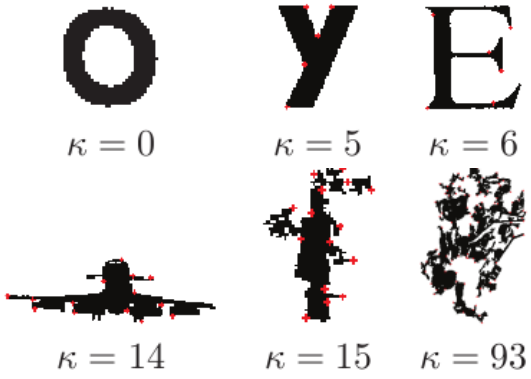
Text Detection via Manually Selected Features

- **Histograms of oriented gradients (Wang et al, 2011)**
  - **Locate characters via computer vision techniques & words via lexicon**
- **Stroke widths (Epshtein et al, 2010)**
  - **Assume characters in images are formed from bands with nearly constant widths**
- **Extremal region properties (Neumann & Matas, 2012)**
  - **Compute region descriptors such as Euler number, horizontal crossings & boundary inflection points**

Stroke width transform

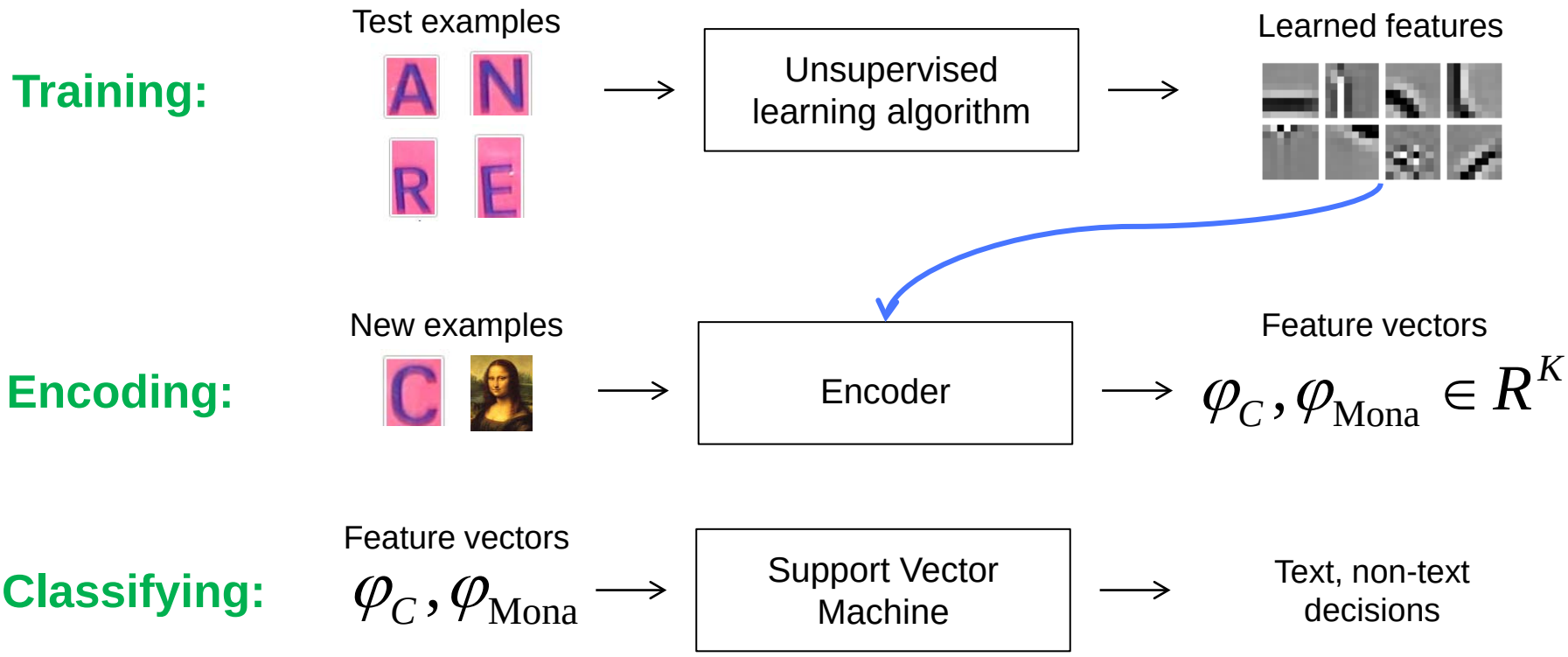


Contour inflection point analysis



Text Detection via Unsupervised Feature Learning

- **Coates et al (2011) advocate learning salient features directly from data instead of handcrafting features**



- **Expensive sliding window used to apply classifier to test images**
  - **Text location & scale determined by brute force**

Prior Art Performance Comparison

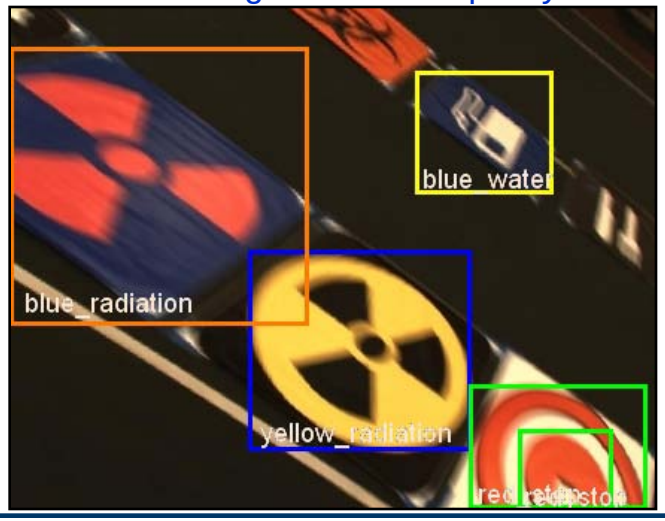
	Tensor voting (2010)	HOG features (2010)	Unsupervised learning (2011)	Extremal regions (2012)
Precision	81% (chars)	75% (words)	60% (chars)	37% (words)
Recall	83% (chars)	25% (words)	30% (chars)	37% (words)
Text recognition	✗	✔	✔	✔
Generality	Horizontal lines/curves	Lexicon-dependent	Automatic features	Hand-crafted features
Speed	?	?	Sliding window	“Real time”

2012 Tech Office Challenge

- **Automatically recognize 9 colored symbols placed at random locations in maze**
  - ***A priori* unknown viewing geometries, illumination conditions & background clutter rendered this constrained problem highly nontrivial**
- **Combined color analysis, extremal region shape properties & unsupervised feature learning to identify signs on a laptop in under 10 secs**
- **Algorithms & computer codes developed for TOC12 can be adapted to more general text recognition problem**

TOC12 symbol found within cluttered scene

Automatic recognition of multiple symbols



Outline

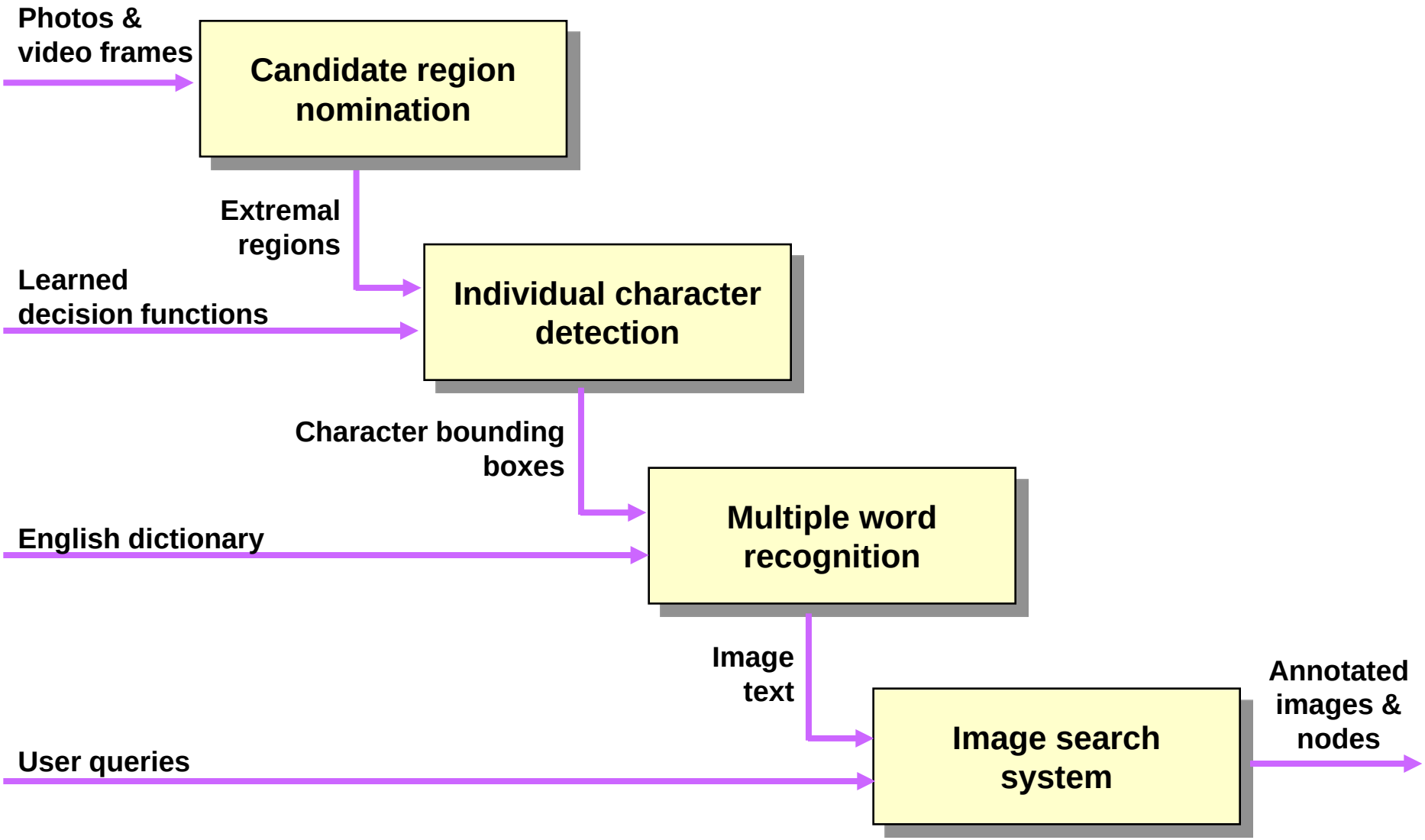
- Prior art
- **Program plan**
- Schedule, budget & follow-on potential

Program Plan Overview

- **Basic objective: Develop imagery text search system that recognizes words in megapixel-sized pictures at a rate exceeding one image per minute on a laptop**

- **Primary tasks**
  - **Work with photos & video clips from internet sites such as Flickr & YouTube**
  - **Nominate candidate character image regions via connected component shape analysis**
  - **Detect individual characters via unsupervised feature learning classifiers trained on synthesized text inputs**
  - **Recognize multiple words after imposing color, image orientation & language model consistency constraints**
  - **Quantify text detection & recognition performance on standard truthed sets**
  - **Integrate text recognition into Image Search System**

Imagery Text Search System



Character Region Nomination

- **Identify connected components that are locally brighter/darker than their immediate surroundings**
  - **Set of all such extremal regions as a function of image binary threshold forms a tree**

Internet photo containing road sign text

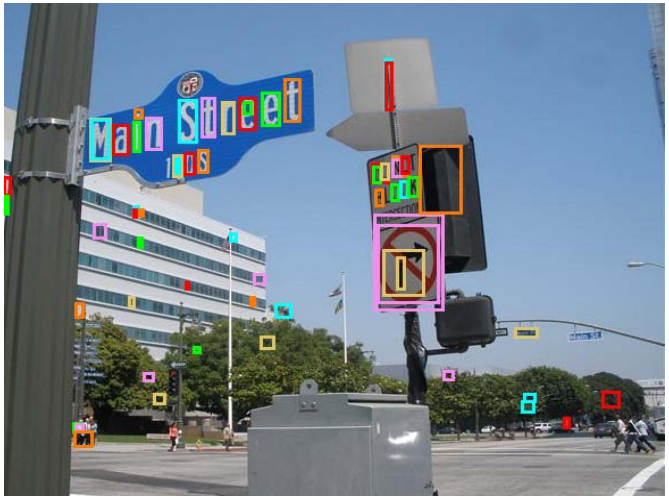
Bright & dark extremal regions computed for particular binary threshold values

Character Region Nomination

- **Identify connected components that are locally brighter/darker than their immediate surroundings**
  - **Set of all such extremal regions as a function of image binary threshold forms a tree**
- **Iteratively evaluate shape properties for each extremal region**
  - **Reject candidates whose aspect ratios, compactness and/or median horizontal crossings significantly disagree with those for text characters**
- **Require candidate regions to remain stable for modest changes in binary image threshold**

Internet photo containing road sign text

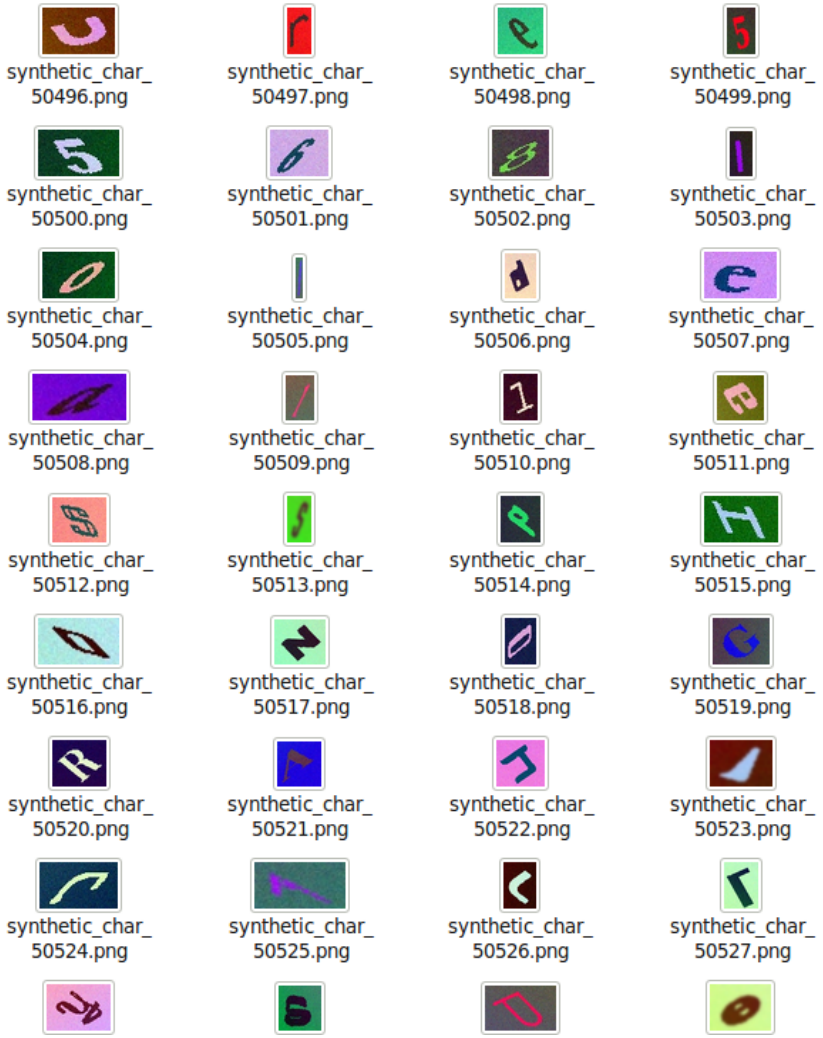
Nominated regions containing individual characters



Synthesizing Text Training Data

- **Large training sets incorporating expected variability in test data are needed for supervised & unsupervised learning methods**
  - **Existing labeled sets of image text are relatively small & homogeneous**
- **Generate 32×32 pictures of characters whose repetition frequencies are set by English word lists**
  - **Randomly convert some letters into numerical digits**
  - **Render characters in 155 different fonts**
- **Introduce variation into synthetic character images via 3D rotations, foreground/background colors, linear shading, gaussian noise & blurring**

Synthesized character images



Individual Character Detection

- **Randomly extract 8×8 pixel patches from synthesized character images**
  - **Whiten each patch by subtracting descriptors' mean & multiplying by inverse square root covariance matrix**
- **Initially assign each patch to one of K=1024 random clusters**
  - **Iteratively update K clusters until dictionary converges**
- **Use dictionary to convert 8×8 patches from text & non-text images into pooled 9K dimensional feature vectors**
- **Generate character decision functions from feature vectors via linear SVM**

Dictionary elements generated from synthetic characters

Characters & false alarms found in internet photo

Multiple Word Recognition

- **Use individual character orientations & sizes to search for words containing at least 3 letters**
  - **Spatial correlations among genuine words within image planes should enable recovery of letters missed at character detection stage**
- **Require gross consistency between foreground & background for all letters within words**
  - **Strings of characters with similar colorings & sizes likely correspond to genuine words**
- **Employ simple spelling & language models to correct inevitable recognition errors**

Notional rejections of spatially isolated "characters" & spatially inconsistent "words"

Notional recovery of missed character

Quantifying Text Detection & Recognition Performance

- **Work with standard truthed image sets (e.g. ICDAR 2003, Street View Text 2011)**

- **First measure character vs non-character detection per image**
  - **Declare character detected if 50% of its bounding box overlaps truth**

Notional character detection evaluation for an ICDAR 03 photo

Quantifying Text Detection & Recognition Performance

- **Work with standard truthed image sets (e.g. ICDAR 2003, Street View Text 2011)**
- **First measure character vs non-character detection per image**
  - **Declare character detected if 50% of its bounding box overlaps truth**
- **Evaluate character recognition via precision & recall metrics**
  - **Precision = $n_{correct}/n_{detected}$**
  - **Recall = $n_{correct}/n_{actual}$**

Notional character recognition evaluation

Character recognition precision=3/5

Character recognition recall=3/5

Quantifying Text Detection & Recognition Performance

- **Work with standard truthed image sets (e.g. ICDAR 2003, Street View Text 2011)**
- **First measure character vs non-character detection per image**
  - **Declare character detected if 50% of its bounding box overlaps truth**
- **Evaluate character recognition via precision & recall metrics**
  - **Precision = $n_{correct}/n_{detected}$**
  - **Recall = $n_{correct}/n_{actual}$**
- **Score word recognition by counting number of reported words with at least 75% correctly spelled characters**

Notional word recognition evaluation

TEscA ✔

value ✔

WaSeR ✕

liqud ✔

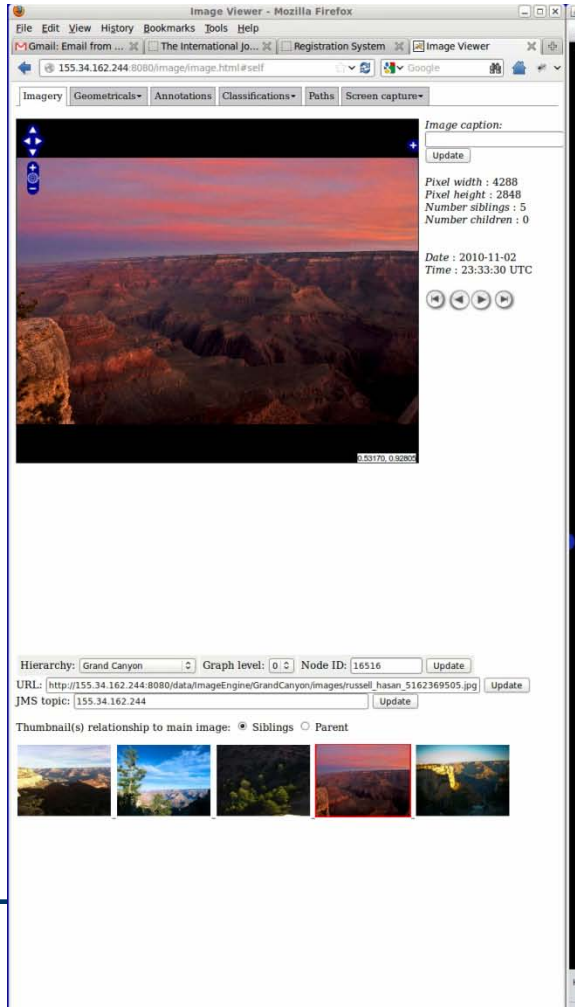
Word recognition precision=3/4

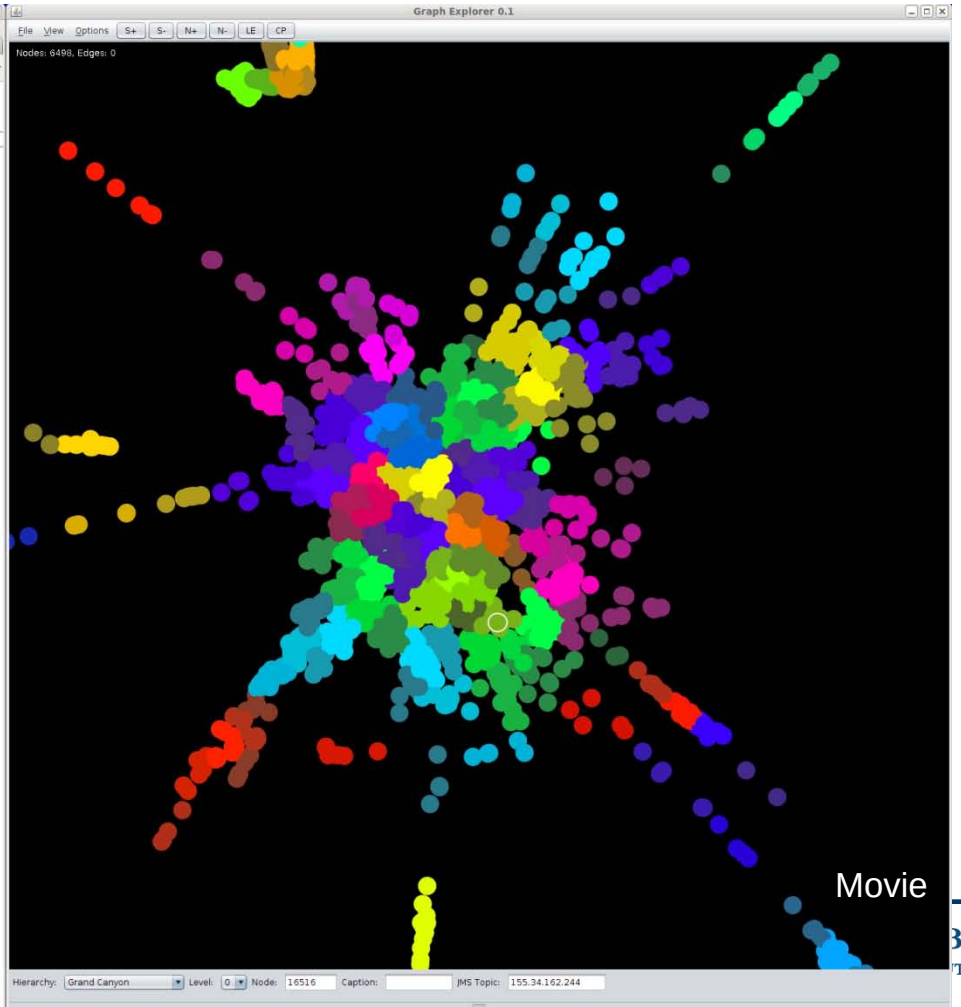
Word recognition recall=3/5

Integrating Text Recognition into Image Search System

- LL tools developed from 2010-12 enable user exploration of O(10^4) images
- Pictures with particular attributes are highlightable in graph viewer
- System can incorporate text querying once it becomes sufficiently robust

Synchronized web browser & graph viewer exploration of 4K+ Flickr photos labeled as "Grand Canyon"

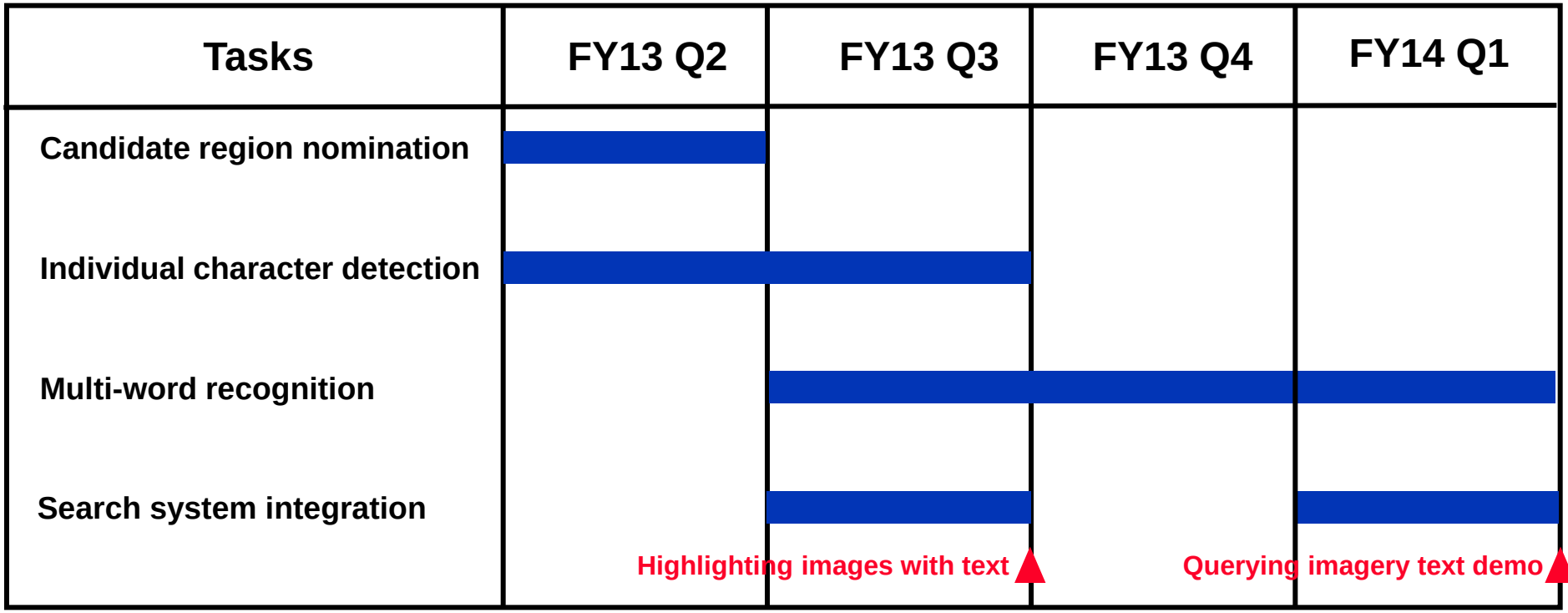




Outline

- Prior art
- Program plan
- **Schedule, budget & follow-on potential**

Schedule & Budget



- **Budget request**
  - **IOE: $90K**
  - **OP: $15K (travel, computer equipment)**
  - **Total: $105K**

Follow-On Potential

FMV analysis (RCO)

UAV frame downlaoded from YouTube

Media monitoring (CIA)

Image geofingerprinting (NGA)

Open source intelligence (Air Force/A2DS)