Joint Visual-Text Modeling for Multimedia Retrieval

JHU CLSP Workshop 2004 - Final Presentation, August 17 2004
Team

- **Undergraduate Students**
  - Desislava Petkova (Mt. Holyoke), Matthew Krause (Georgetown)

- **Graduate Students**
  - Shaolei Feng (U. Mass), Brock Pytlik (JHU), Paola Virga (JHU)

- **Senior Researchers**
  - Pinar Duygulu, Bilkent U., Turkey
  - Pavel Ircing (U. West Bohemia)
  - Giri Iyengar, IBM Research
  - Sanjeev Khudanpur, CLSP, JHU
  - Dietrich Klakow, Uni. Saarland
  - R. Manmatha, CIIR, U. Mass Amherst
  - Harriet Nock, IBM Research (external participant)
Big Picture: Multimedia Retrieval Task

Most research has addressed:
I. Text queries, text (or degraded text) documents
II. Image queries, image data

Joint-Visual Text Models!
Joint Visual-Text Modeling

Document of words → Query of words → Document of Words and Visterms → Query of Words and Visterms

Retrieve documents using $p(\text{Document} | \text{Query})$

Retrieve documents using $p(\text{dw}, \text{dv} | \text{qw}, \text{qv})$

Yasser Arafat

Process Query Text

Process Query Image

Joint visterm retrieval

VIDEO CLIPS

“... [Yes sir, you’re fat today said]..."
Joint Visual-Text Modeling: KEY GOAL

- Show that joint visual-text modeling improves multimedia retrieval
  - Demonstrate and Evaluate performance of these models on TRECVID2003 corpus and task
Key Steps

- Automatically annotate video with concepts (meta-data)
  - E.g. Video contains a face, in a studio-environment ...

- Retrieve video
  - Given a query, select suitable meta-data for the query and retrieve
  - Combine with text-retrieval in a unified Language Model-based IR setting
TRECVID Corpus and Task

- **Corpus**
  - Broadcast news videos used for Hub4 evaluations (ABC, CNN, CSPAN)
  - 120 Hours of video

- **Tasks**
  - Shot-boundary detection
  - News Story segmentation (multimodal)
  - Concept detection (Annotation)
  - Search task
Alternate (development) Corpus

- COREL photograph database
  - 5000 high-quality photographs with captions
- Task
  - Annotation
TRECVID Search task definition

NIST Evaluation

Statement of Information need + Examples

Manual Selection of System Parameters

Ranked list of video shots

Manual Interactive
Our search task definition

NIST Evaluation

Statement of Information need + Examples

Automatic Selection of System Parameters

Ranked list of video shots

Isolate Algorithmic issues from interface and user issues
Rank documents with $p(q_w,q_v|d_w,d_v)$

Baseline model

Relating document visterms to query words (MT, Relevance Model, HMMs)

Relating document words to query images (Text Classification experiments)

Visual-only retrieval models
Evaluation

- Concept annotation performance
  - Compare against manual ground truth

- Retrieval task performance
  - Compare against NIST relevance judgements

- Both measured using Mean Average Precision (mAP)
Mean Average Precision (mAP)

\[ S(t) = \sum_{i \in \{\text{relevant}\}} \text{precision}(i) \]

\[ AP(t) = \frac{S(t)}{|\text{rel}(t)|} \]

\[ mAP = \frac{\sum_{t \in T} AP(t)}{|T|} \]
Experimental Setup: Corpora

**TRECVID03 Corpus**
- 120 Hours
- Ground Truth on Dev data

**Train**
- 38K shots

**Dev**
- Test 10K shots

**TRECVID03 IR Collection**
- 32K Shots

**COREL Corpus**
- 5000 images

**Train**
- 4500 images

**Test**
- 500 images
Experimental Setup: Visual Features

Original

L*a*b  Edge Strength  Co-occurrence
Interest Point Neighborhoods (Harris detector)

Greyscale image

Interest points

points detected
Experimental Setup: Visual Feature list

- Regular partition
  - L*a*b Moments (COLOR)
  - Smoothed Edge Orientation Histogram (EDGE)
  - Grey-level Co-occurrence matrix (TEXTURE)

- Interest Point neighborhood
  - COLOR, EDGE, TEXTURE
Presentation Outline

Translation (MT) models (Paola),

Relevance Models (Shao Lei, Desislava),

Graphical Models (Pavel, Brock)

Text classification models (Matt)

Integration & Summary (Dietrich)
A Machine Translation Approach to Image Annotation

Presented by Paola Virga
Translation (MT) models

\[ p(q_w \mid d_v) = \sum_c p(q_w \mid c) p(c \mid d_v) \]
Inspiration from Machine Translation

\[ p(f|e) = \sum_a p(f,a|e) \]
\[ p(c|v) = \sum_a p(c,a|v) \]

Direct translation model
Discrete Representation of Image Regions (visterms) to create analogy to MT

In Machine Translation $\rightarrow$ discrete tokens

In our task

However, the features extracted from regions are continuous

Solution: Vector quantization $\rightarrow$ visterms $\checkmark$

$\{f_{n1}, f_{n2}, \ldots f_{nm}\} \rightarrow v_k$
Image annotation using translation probabilities

\[ p(c|v) : \text{Probabilities obtained from direct translation} \quad p(\text{sun | \ }) \]

\[ P_0(c \mid d_v) = \frac{1}{|d_v|} \sum_{v \in d_v} P(c \mid v) \]
Annotation Results (Corel set)

Top: manual annotations, bottom: predicted words (top 5 words with the highest probability)
Red: correct matches
Feature selection

Features: color, texture, edge
Extracted from blocks, or around interest points

Observations

- Features extracted from blocks give better performance than features extracted around interest points
- When the features are used individually, edge features give the best performance
- Training using all is the best
  - Using Information Gain to select visterms vocabulary didn’t help
- Integrating number of faces, increases the performance slightly

mAP values for different features

<table>
<thead>
<tr>
<th>Feature</th>
<th>mAP Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Color</td>
<td>0.12</td>
</tr>
<tr>
<td>Texture</td>
<td>0.10</td>
</tr>
<tr>
<td>Edge</td>
<td>0.14</td>
</tr>
<tr>
<td>All</td>
<td>0.16</td>
</tr>
<tr>
<td>All + face</td>
<td>0.18</td>
</tr>
</tbody>
</table>
Model and iteration selection

Strategies compared
(a) IBM Model 1 $p(c|v)$
(b) HMM on top of (a)
(c) IBM Model 4 on top of (b)

-> Observation: IBM Model 1 is the best

<table>
<thead>
<tr>
<th>Corel</th>
<th>TREC</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.125</td>
<td>0.124</td>
</tr>
</tbody>
</table>

Number of iterations in Giza training affects the performance
-> Less iterations give better annotation performance but cannot produce rare words
Integrating word co-occurrences

- Model 1 with word co-occurrence

\[ P_1(c_i \mid d_v) = \sum_{j=1}^{\lfloor C \rfloor} P(c_i \mid c_j) P_0(c_j \mid d_v) \]

- Integrating word co-occurrences into the model helps for Corel but not for TREC

<table>
<thead>
<tr>
<th></th>
<th>Corel</th>
<th>TREC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model 1</td>
<td>0.125</td>
<td>0.124</td>
</tr>
<tr>
<td>Model 1 + Word-CO</td>
<td>0.145</td>
<td>0.124</td>
</tr>
</tbody>
</table>
Inspiration from CLIR

- Treat Image Annotation as a Cross-lingual IR problem
  - Visual Document comprising visterms (target language) and a query comprising a concept (source language)

\[
p(c \mid d_v) = \lambda \left( \sum_{v \in V} p(c \mid v)p(v \mid d_v) \right) + \left( 1 - \frac{\lambda}{4} \right) p(c \mid G_3) \quad \text{same} \forall d_v
\]
Inspiration from CLIR

- Treat Image Annotation as a Cross-lingual IR problem
  - Visual Document comprising visterms (target language) and a query comprising a concept (source language)

\[
p(c \mid d_v) = \sum_{v \in d_v} p(v \mid d_v) p(c \mid v)
\]

- Image does not provide a good estimate of \(p(v \mid d_v)\)
- Tried \(p(v)\) and \(DF(v)\), DF works best

\[
\text{score}(c \mid d_v) = \sum_{v \in d_v} DF_{\text{Train}}(v) p(c \mid v)
\]
# Annotation Performance on TREC

<table>
<thead>
<tr>
<th>Model 1</th>
<th>0.124</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CLIR using Model 1</strong></td>
<td>0.126</td>
</tr>
</tbody>
</table>

Significant at p=0.04

Average Precision values for the top 10 words:
For some concepts we achieved up to 0.6
Annotation Performance on TREC
Questions?
Relevance Models for Image Annotation

Presented by Shaolei Feng
University of Massachusetts, Amherst
Relevance Models as Visual Model

Use Relevance Models to estimate the probabilities of concepts given test keyframes.

\[
p(q_w \mid d_v) = \sum_c p(q_w \mid c)p(c \mid d_v)
\]
Intuition

- Images are defined by spatial context.
  - Isolated pixels have no meaning.
  - Context simplifies recognition/retrieval.
  - E.g. Tiger is associated with grass, tree, water, forest.
  - Less likely to be associated with computers.
Introduction to Relevance Models

- Originally introduced for text retrieval and cross-lingual retrieval
  - Lavrenko and Croft 2001, Lavrenko, Choquette and Croft, 2002
  - A formal approach to query expansion.

- A nice way of introducing context in images
  - Without having to do this explicitly
  - Do this by computing the joint probability of images and words
Cross Media Relevance Models (CMRM)

- Two parallel vocabularies: Words and Visterms
- Analogous to Cross-lingual relevance models
- Estimate the joint probabilities of words and visterms from training images

\[ P(c, d_v) = \sum_{J \in T} P(J)P(c \mid J) \prod_{i=1}^{d_v} P(v_i \mid J) \]

Continuous Relevance Models (CRM)

- A continuous version of Cross Media Relevance Model
- Estimate the $P(v|J)$ using kernel density estimate

\[
P(v | J) = \frac{1}{n} \sum_{i=1}^{J} K\left(\frac{\|v - v_{Ji}\|}{\beta}\right)
\]

$K$: Gaussian Kernel
$\beta$: Bandwidth
Continuous Relevance Model

- A generative model
- Concept words $w_j$ generated by an i.i.d. sample from a multinomial
- Visterms $v_i$ generated by a multi-variate (Gaussian) density
Normalized Continuous Relevance Models

- Normalized CRM
  - Pad annotations to fixed length. Then use the CRM.
  - Similar to using a Bernoulli model (rather than a multinomial for words).
  - Accounts for length (similar to length of document in text retrieval).

S. L. Feng, V. Lavrenko and R. Manmatha, *Multiple Bernoulli Models for Image and Video Annotation*, in CVPR’04
V. Lavrenko, S. L. Feng and R. Manmatha, *Statistical Models for Automatic Video Annotation and Retrieval*, in ICASSP04
Annotation Performance

- On Corel data Set:

<table>
<thead>
<tr>
<th>Models</th>
<th>CMRM</th>
<th>CRM</th>
<th>Normalized-CRM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean average Precision</td>
<td>0.14</td>
<td>0.23</td>
<td>0.26</td>
</tr>
</tbody>
</table>

- Normalized-CRM works best
Annotation Examples (Corel set)

- **Sky** train railroad locomotive water
- **Cat** tiger bengal tree forest
- **Snow fox** arctic tails water
- **Tree** plane zebra herd water
- **Birds** leaf nest water sky
- **Mountain** plane jet water sky
Results: Relevance Model on Trec Video Set

- **Model**: Normalized continuous relevance model
- **Features**: color and texture
  - Our comparison experiments show adding edge feature only get very slight improvement
- **Evaluate annotation on the development dataset for annotation evaluation**
  - mean average precision: 0.158
Annotation Performance on TREC
Proposal: Using Dynamic Information for Video Retrieval

Presented by Shaolei Feng
University of Massachusetts, Amherst
Motivation

- Current models based on single frames in each shot.
- But video is dynamic
  - Has motion information.
- Use dynamic (motion) information
  - Better image representations (segmentations)
  - Model events/actions
Why Dynamic Information

- Model actions/events
  - Many Trecvid 2003 queries require motion information. E.g.
    - find shots of an airplane *taking off*.
    - find shots of a person *diving into* water.
  - Motion is an important cue for retrieving actions/events.
    - But using the optical flow over the entire image doesn't help.
    - Use motion features from objects.

- Better Image Representations
  - Much easier to segment moving objects from background than to segment static images.
Problems with still images.

- **Current approach**
  - Retrieve videos using static frames.

- **Feature representations**
  - Visterms from keyframes.
  - Rectangular partition or static segmentation
    - Poorly correlated with objects.
  - Features – color, texture, edges.

- Problem: visterms not correlated well with concepts.
Better Visterms - better results.

- Model performs well on related tasks.
- Retrieval of handwritten manuscripts.
  - Visterms - word images.
  - Features computed over word images.
  - Annotations - ASCII word.
    - “you are to be particularly careful”
  - Segmentation of words easier.
  - Visterms better correlated with concepts.
- So can we extend the analogy to this domain…
Segmentation Comparison

a: Segmentation using only still image information

b: Segmentation using only motion information

Pictures from Patrick Bouthemy’s Website, INRIA
Represent Shots not Keyframes

- Shot boundary detection
  - Use standard techniques.

- Segment moving objects
  - E.g. By finding outliers from dominant (camera) motion.

- Visual features for object and background.

- Motion features for object
  - E.g. Trajectory information,

- Motion features for background.
  - Camera pan, zoom ...
Models

- One approach - modify relevance model to include motion information.
- Probabilistically annotate shots in the test set.

\[
P(c, d_v) = \sum_{J \in T} P(J)P(c | J) \prod_{i=1}^{d_v} P(v_i | J)
\]

\[
P(c, (d_v, d_m)) = \sum_{S \in T} P(S)P(c | S) \prod_{i=1}^{d} P(v_i | S)P(m_i | S)
\]

T: training set, S: shots in the training set

- Other models e.g. HMM also possible
Estimation $P(v_i | S), P(m_i | S)$

- If discrete visit terms use smoothed maximum likelihood estimates.
- If continuous use kernel density estimates.
- Take advantage of repeated instances of the same object in shot.
Plan

- Modify models to include dynamic information
- Train on TrecVID03 development dataset
- Test on TrecVID03 test dataset
  - Annotate the test set
  - Retrieve using TrecVID 2003 queries.
  - Evaluate retrieval performance using mean average precision
Score Normalization
Experiments

Presented by Desislava Petkova
Motivation for Score Normalization

- Score probabilities are small
- But there seems to be discriminating power
- Try to use likelihood ratios
Bayes Optimal Decision Rule

\[ r(s) = \frac{P(w|s)}{P(\bar{w}|s)} = \frac{P(s)P(w|s)}{P(s)P(\bar{w}|s)} = \frac{P(w)P(s|w)}{P(\bar{w})P(s|\bar{w})} = \frac{p(w) \text{pdf}_w(s|w)}{p(\bar{w}) \text{pdf}_\bar{w}(s|\bar{w})} \]

\[ P(w|s) = \frac{r(s)}{1+r(s)} \]
Estimating Class-Conditional PDFs

- For each word:
  - Divide training images into positive and negative examples
  - Create a model to describe the score distribution of each set
    - Gamma
    - Beta
    - Normal
    - Lognormal

- Revise word probabilities
Annotation Performance

- Did not improve annotation performance on Corel or TREC
Proposal: Using Clustering to Improve Concept Annotation

Desislava Petkova
Mount Holyoke College
17 August 2004
Automatically annotating images

- **Corel:**
  - 5000 images
    - 4500 training
    - 500 testing
- **Word vocabulary**
  - 374 words
- **Annotations**
  - 1-5 words
- **Image vocabulary**
  - 500 visterms
Relevance models for annotation

- A generative language modeling approach
- For a test image $I = \{v_1, ..., v_m\}$ compute the joint distribution of each word $w$ in the vocabulary with the visterms of $I$
  - Compare $I$ with training images $J$ annotated with $w$

$$P(w, I) = \sum_{J \in T} P(J)P(w, I|J)$$

$$= \sum_{J \in T} P(J)P(w|J) \prod_{i=1}^{m} P(v_i|J)$$
Estimating $P(w|J)$ and $P(v|J)$

- Use maximum-likelihood estimates
  - Smooth with the entire training set $T$

\[
P(w|J) = (1 - a) \frac{c(w, J)}{|J|} + a \frac{c(w, T)}{|T|}
\]

\[
P(v|J) = (1 - b) \frac{c(v, J)}{|J|} + b \frac{c(v, T)}{|T|}
\]
Motivation

- Estimating the relevance model of a single image is a noisy process
  - $P(v|J)$: visterm distributions are sparse
  - $P(w|J)$: human annotations are incomplete
- Use clustering to get better estimates
Potential benefits of clustering

Words in red are missing in the annotation
Relevance Models with Clustering

- Cluster the training images using K-means
  - Use both visterms and annotations
- Compute the joint distribution of visterms and words in each cluster
  - Use clusters instead of individual images

\[
P(w, I) = \sum_{C \in T} P(C) P(w|C) \prod_{i=1}^{m} P(v_i|C)
\]
Preliminary results on annotation performance

<table>
<thead>
<tr>
<th>Model Description</th>
<th>mAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard relevance model (4500 training examples)</td>
<td>0.14</td>
</tr>
<tr>
<td>Relevance model with clusters (100 training examples)</td>
<td>0.128</td>
</tr>
</tbody>
</table>
Cluster-based smoothing

- Smooth maximum likelihood estimates for the training images based on clusters they belong to

\[
P(w|J) = (1-a_1-a_2) \frac{c(w, J)}{|J|} + a_1 \frac{c(w, C_J)}{|C_J|} + a_2 \frac{c(w, T)}{|T|}
\]

\[
P(v|J) = (1-b_1-b_2) \frac{c(v, J)}{|J|} + b_1 \frac{c(v, C_J)}{|C_J|} + b_2 \frac{c(v, T)}{|T|}
\]
Experiments

- Optimize smoothing parameters
  - Divide training set
    - 4000 training images
    - 500 validation images
- Find the best set of clusters
  - Query-dependent clusters
  - Investigate soft clustering
Evaluation plan

- Retrieval performance
  - Average precision and recall for one-word queries
  - Comparison with the standard relevance model
Hidden Markov Models for Image Annotations

Pavel Ircing
Sanjeev Khudanpur
Presentation Outline

Translation (MT) models
(Paola),

Relevance Models
(Shao Lei, Desislava),

Graphical Models
(Pavel, Brock)

Text classification models
(Matt)

Integration & Summary
(Dietrich)
Model setup

Training HMMs:
- separate HMM for each training image - states given by manual annotations.
- image blocks are “generated” by annotation words
- alignment between image blocks and annotation words is a hidden variable, models are trained using the EM algorithm (HTK toolkit)

Test HMM has \(|W|\) states, 2 scenarios:
(a) \(p(w'|w)\) uniform
(b) \(p(w'|w)\) from co-occurrence LM

Posterior probability from forward-backward pass used for \(p(w|\text{Image})\)
Challenges in HMM training

- Inadequate annotations
- There is no notion of order in the annotation words
  - Difficulties with automatic alignment between words and image regions
- No linear order in image blocks (assume raster-scan)
  - Additional spatial dependence between block-labels is missed
  - Partially addressed via a more complex DBN (see later)
Inadequacy of the annotations

- Corel database
  - Annotators often mark only interesting objects

- TRECVID database
  - Annotation concepts capture mostly semantics of the image and they are not very suitable for describing visual properties
Alignment problems

- There is no notion of order in the annotation words
  - Difficulties with automatic alignment between words and image regions
Gradual Training

- Identify a set of “background” words (sky, grass, water,...)
- In the initial stages of HMM training
  - Allow only “background” states to have their individual emission probability distributions
  - All other objects share a single “foreground” distribution
- Run several EM iterations
- Gradually untie the “foreground” distribution and run more EM iterations
Gradual Training Results

**Results:**
- Improved alignment of training images
- Annotation performance on test images did not change significantly
Another training scenarios

- models were forced to visit every state during training
  - huge models, marginal difference in performance

- special states introduced to account for unlabelled background and unlabelled foreground, with different strategies for parameter tying
## Annotation performance - Corel

<table>
<thead>
<tr>
<th>Image features</th>
<th>LM</th>
<th>mAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discrete</td>
<td>No</td>
<td>0.120</td>
</tr>
<tr>
<td></td>
<td>Yes</td>
<td>0.150</td>
</tr>
<tr>
<td>Continuous (1 Gaussian per state)</td>
<td>No</td>
<td>0.140</td>
</tr>
<tr>
<td></td>
<td>Yes</td>
<td>0.157</td>
</tr>
</tbody>
</table>

- Continuous features are better than discrete
- Co-ocurrence language model also gives moderate improvement
### Annotation performance - TRECVID

- Continuous features only, no language model

<table>
<thead>
<tr>
<th>Model</th>
<th>LM</th>
<th>mAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Gaussian per state</td>
<td>No</td>
<td>0.094</td>
</tr>
<tr>
<td></td>
<td>Yes</td>
<td>X</td>
</tr>
<tr>
<td>12 Gaussians per state</td>
<td>No</td>
<td>0.145</td>
</tr>
<tr>
<td></td>
<td>Yes</td>
<td>X</td>
</tr>
</tbody>
</table>
Annotation Performance on TREC
Summary: HMM-Based Annotation

- **Very encouraging preliminary results**
  - Effort started this summer, validated on Corel, and yielded competitive annotation results on TREC

- **Initial findings**
  - Proper normalization of the features is crucial for system performance: bug found and fixed on Friday!
  - Simple HMMs seem to work best
    - More complex training topology didn't really help
    - More complex parameter tying was only marginally helpful

- **Glaring gaps**
  - Need a good way to incorporate a language model
Graphical Models for Image Annotation
+
Joint Segmentation and Labeling for Content Based Image Retrieval

Brock Pytlik
Johns Hopkins University
bep@cs.jhu.edu
Outline

- Graphical Models for Image Annotation
  - Hidden Markov Models
    - Preliminary Results
  - Two-Dimensional HMM’s
    - Work in Progress

- Joint Image Segmentation and Labeling
  - Tree Structure Models of Image Segmentation
    - Proposed Research
Graphical Model Notation

$C_1 \quad p(c \mid c') \quad C_2 \quad p(c \mid c') \quad C_3$

$\begin{align*}
\text{water} & \quad \text{grass} & \quad \text{tiger} \\
\text{ground} & \quad \text{water} & \quad \text{ground} & \quad \text{tiger} \\
\text{grass} & \quad \text{water} & \quad \text{grass} & \quad \text{tiger} \\
\text{tiger} & \quad \text{water} & \quad \text{ground} & \quad \text{tiger} \\
\end{align*}$

$p(o \mid c) \quad p(o \mid c) \quad p(o \mid c)$

$O_1 \quad O_2 \quad O_3$

Images of tiger and grass with water symbol.
Graphical Model Notation

\[ C_1 \rightarrow p(c | c') \rightarrow C_2 \rightarrow p(c | c') \rightarrow C_3 \]

\[ p(o | c) \]

\[ O_1 \rightarrow O_2 \rightarrow O_3 \]
Graphical Model Notation

\[
\begin{align*}
C_1 & \quad p(c | c') \\
C_2 & \quad p(c | c') \\
C_3 & \quad p(o | c) \\
O_1 & \quad p(o | c) \\
O_2 & \quad p(o | c) \\
O_3 & \quad p(o | c)
\end{align*}
\]
Graphical Model Notation
Graphical Model Notation Simplified

An HMM for a 24-block Image
Graphical Model Notation Simplified

An HMM for a 24-block Image
Modeling Spatial Structure

An HMM for a 24-block Image
Modeling Spatial Structure

An HMM for a 24-block Image

Transition probabilities represent spatial extent of objects
Modeling Spatial Structure

Transition probabilities represent spatial extent of objects

A Two-Dimensional Model for a 24-block Image
Modeling Spatial Structure

Transition probabilities represent spatial extent of objects

A Two-Dimensional Model for a 24-block Image

<table>
<thead>
<tr>
<th>Model</th>
<th>Training Time Per Image</th>
<th>Training Time Per Iteration</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-D HMM</td>
<td>.5 sec</td>
<td>37.5 min</td>
</tr>
<tr>
<td>2-D HMM</td>
<td>110 sec</td>
<td>8250 min = 137.5 hr</td>
</tr>
</tbody>
</table>
Bag-of-Annotations Training
Unlike ASR Annotation Words are Unordered

\[
p(M_t = 1) = \begin{cases} 
1 & \text{if } c_t \in \{\text{tiger, grass, sky}\} \\
0 & \text{otherwise}
\end{cases}
\]
Bag-of-Annotations Training (II)  
Forcing Annotation Words to Contribute

\[ M_t^{(1)} = M_{t-1}^{(1)} \lor (C_t = \text{tiger}) \]
\[ M_t^{(2)} = M_{t-1}^{(2)} \lor (C_t = \text{grass}) \]
\[ M_t^{(3)} = M_{t-1}^{(3)} \lor (C_t = \text{sky}) \]

Only permit paths that visit every annotation word.
Inference on Test Images

- Forward Decoding

\[ p(c \mid d_v) = \frac{p(c, d_v)}{p(d_v)} \]
Inference on Test Images

- Forward Decoding

\[ p(c \mid d_v) = \frac{p(c, d_v)}{p(d_v)} = \sum_{S \in \mathcal{C}} \left[ \prod_{i=1}^{N} p(v_i \mid s_i) \right] p(S) \]
Inference on Test Images

- Forward Decoding

\[
p(c \mid d_v) = \frac{p(c, d_v)}{p(d_v)} = \frac{\sum_{S \ni c} \left[ \prod_{i=1}^{N} p(v_i \mid s_i) \right] p(S)}{\sum_{S} \left[ \prod_{i=1}^{N} p(v_i \mid s_i) \right] p(S)}
\]
Inference on Test Images

- **Forward Decoding**

\[
p(c \mid d_v) = \frac{p(c, d_v)}{p(d_v)} = \frac{\sum_{S \ni c} \left[ \prod_{i=1}^{N} p(v_i \mid s_i) \right] p(S)}{\sum_{S} \left[ \prod_{i=1}^{N} p(v_i \mid s_i) \right] p(S)}
\]

- **Viterbi Decoding**
  - **Approximate Sum over all Paths with the Best Path**
# Annotation Performance on Corel Data

<table>
<thead>
<tr>
<th>Model</th>
<th>Image Features</th>
<th>mAP</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Discrete</td>
<td>0.071</td>
</tr>
<tr>
<td></td>
<td>Continuous</td>
<td>0.086</td>
</tr>
<tr>
<td></td>
<td>Continuous</td>
<td>0.074</td>
</tr>
</tbody>
</table>

- Working with 2-D models needs further study
- mAP not yet on par with other models
Future Work

- Improved Training for Two-Dimensional Models
  \[ p(c_{i,j} \mid c_{i-1,j}, c_{i,j-1}) \propto p(c \mid c_{i-1,j}) p(c \mid c_{i-1,j}) \]
  - Permits training horizontal and vertical chains separately
- Other variations could be investigated

- Next Idea
  - Joint Image Segmentation and Labeling
Joint Segmentation and Labeling

tiger, grass, sky
Joint Segmentation and Labeling

tiger, grass, sky
Joint Segmentation and Labeling

tiger, grass, sky
Joint Segmentation and Labeling

tiger, grass, sky
Research Proposal

- A Generative Model for Joint Segmentation and Labeling
  - Tree construction by agglomerative clustering of image regions (blocks) based on visual similarity
    - Segmentation = A cut across the resulting tree
    - Labeling = Assigning concepts to resulting leaves
Model

- General Model

\[ p(\bar{c}, d_v) = \sum_{u \in \text{cuts}(\text{tree}(d_v))} p(u) \prod_{l \in \text{leaves}(u)} p(c_l \mid u, l) p(\text{obs}(l) \mid c_l) \]
Model

General Model

\[ p(\overline{c}, d_v) = \sum_{u \in \text{cuts(tree}(d_v))} p(u) \prod_{l \in \text{leaves}(u)} p(c_l | u, l) p(\text{obs}(l) | c_l) \]

Probability of Cut
Model

- **General Model**

\[ p(\overline{c}, d_v) = \sum_{u \in \text{cuts}(\text{tree}(d_v))} p(u) \prod_{l \in \text{leaves}(u)} p(c_l | u, l) p(\text{obs}(l) | c_l) \]

Probability of Label Given Cut and Leaf
Model

- **General Model**

\[ p(\bar{c}, d_v) = \sum_{u \in \text{cuts}(\text{tree}(d_v))} p(u) \prod_{l \in \text{leaves}(u)} p(c_l | u, l) p(\text{obs}(l) | c_l) \]

- Probability of Observation Given Label
Model

- **General Model**

\[
p(\vec{c}, d_v) = \sum_{u \in \text{cuts}(\text{tree}(d_v))} p(u) \prod_{l \in \text{leaves}(u)} p(c_l | u, l) p(\text{obs}(l) | c_l)
\]

- **Independent Generation of Observations Given Label**

\[
p(\vec{c}, d_v) = \sum_{u \in \text{cuts}(\text{tree}(d_v))} p(u) \prod_{l \in \text{leaves}(u)} p(c_l | u, l) \prod_{o \in \text{child}(l)} p(o | c_l)
\]
Estimating Model Parameters

- Suitable independence assumptions may need to be made
  - All cuts are equally likely?
  - Given a cut, leaf labels have a Markov dependence
  - Given a label, its image footprint is independent neighboring image regions

- Work out EM algorithm for this model
Estimating Cuts given Topology

- **Uniform**
  - All cuts containing $|\bar{c}|$ leaves or more equally likely

- **Hypothesize number of segments produced**
  - Hypothesize which possible segmentation used

- **Greedy Choice**
  - Pick node with largest observation probability remaining that produces a valid segmentation
    - Repeat until all observations accounted for

- **Changes Model**
  - No longer distribution over cuts
  - Affects valid labeling strategies
Estimating Labels Given Cuts

- **Uniform**
  - Like HMM training with fixed concept transitions

- **Number of Children**
  - Sky often generates a large number of observations
  - Canoe often generates a small number of observations

- **Co-occurrence Language Model**
  - Eliminates label independence given cut
  - Could do two-pass model like MT group did (not exponential)

\[
p_2(c \mid u, l) = \sum_{a \in C} \left[ \sum_{m \in \text{leaves}(u)} p_1(a \mid m) \right] p(c \mid a)
\]
Estimating Observations Given Labels

- Label Generates its Observations Independently
  - Problem: Product of Children at least as high as Parent Score
- Label Generates Composite Observation at Node
Evaluation Plan

- Evaluate on Corel Image set using mAP
- TREC annotation task
Questions?
Predicting Visual Concepts From Text

Presented by
Matthew Krause
Presentation Outline

Translation (MT) models (Paola),

Relevance Models (Shao Lei, Desislava),

Graphical Models (Pavel, Brock)

Text classification models (Matt)

Integration & Summary (Dietrich)
A Motivating Example
A Motivating Example

<Word stime="177.09" dur="0.22" conf="0.727"> IT'S </Word>
<Word stime="177.31" dur="0.25" conf="0.963"> MUCH </Word>
<Word stime="177.56" dur="0.11" conf="0.976"> THE </Word>
<Word stime="177.67" dur="0.29" conf="0.977"> SAME </Word>
<Word stime="177.96" dur="0.14" conf="0.980"> IN </Word>
<Word stime="178.10" dur="0.13" conf="0.603"> THE </Word>
<Word stime="178.38" dur="0.57" conf="0.953"> SUMMERTIME </Word>
<Word stime="178.95" dur="0.50" conf="0.976"> GLACIER </Word>
<Word stime="179.45" dur="0.60" conf="0.974"> AVALANCHE </Word>
Concepts

- Assume there is a hidden variable $c$ which generates query words from a document's visterms.

\[
p(q_v \mid d_w) = \sum_{c} p(q_v \mid d_w, c) p(c \mid d_w) \equiv \sum_{c} p(q_v \mid c) p(c \mid d_w)
\]
ASR → Features Example

STEVE FOSSETT AND HIS BALLOON SOLO SPIRIT ARSENIDE OVER THE BLACK SEA DRIFTING SLOWLY TOWARDS THE COAST OF THE CAUCUSES HIS TEAM PLANS IF NECESSARY TO BRING HIM DOWN AFTER DAYLIGHT TOMORROW YOU THE CHECHEN CAPITAL OF GROZNY
Building Features

- Insert Sentence Boundaries
- Case Restoration
- Noun Extraction
- Named Entity Detection
- WordNet Processing
- Feature Set
ASR → Features Example

STEVE FOSSETT AND HIS BALLOON SOLO SPIRIT ARSENIDE OVER THE BLACK SEA DRIFTING SLOWLY TOWARDS THE COAST OF THE CAUCUSES HIS TEAM PLANS IF NECESSARY TO BRING HIM DOWN AFTER DAYLIGHT TOMORROW YOU THE CHECHEN CAPITAL OF GROZNY
ASR → Features Example

STEVE FOSSETT AND HIS BALLOON SOLO SPIRIT ARSENIDE.

OVER THE BLACK SEA DRIFTING SLOWLY TOWARDS THE COAST OF THE CAUCUSES.

HIS TEAM PLANS IF NECESSARY TO BRING HIM DOWN AFTER DAYLIGHT TOMORROW.

YOU THE CHECHEN CAPITAL OF GROZNY
ASR → Features Example

Steve Fossett and his balloon Solo Spirit arsenide.

Over the Black Sea drifting slowly towards the coast of the caucuses.

His team plans if necessary to bring him down after daylight tomorrow.

you the Chechan capital of Grozny....
ASR $\rightarrow$ Features Example

Steve Fossett and his balloon Solo Spirit arsenide.

Over the Black Sea drifting slowly towards the coast of the caucuses.

His team plans if necessary to bring him down after daylight tomorrow.

you the Chechan capital of Grozny.
ASR → Features Example

Steve Fossett and his balloon Solo Spirit arsenide.

Over the Black Sea drifting slowly towards the coast of the caucuses.

His team plans if necessary to bring him down after daylight tomorrow.

you the Chechan capital of Grozny.
ASR → Features Example

Steve Fossett and his balloon Solo Spirit arsenide.

Over the Black Sea drifting slowly towards the coast of the caucuses.

His team plans if necessary to bring him down after daylight tomorrow.

you the Chechan capital of Grozny.
Feature Selection

- Basic feature set (nouns + NEs) has \( \sim 18,000 \) elements/shot
  - 6000 elements \( \times \) \{previous, this, next\}
- Using only a subset of the possible features may affect performance.
- Two strategies for feature selection:
  - Remove very rare words (18,000 \( \rightarrow \) 7902)
  - Eliminate low-value features
Information Gain

- Measures the change in entropy given the value of a single feature

\[
Gain(C, F) = H(C) - \sum_{w \in Values(F)} p(w) H(C \mid F = w)
\]
## Information Gain Results

<table>
<thead>
<tr>
<th>Basketball</th>
<th>Sky</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. (empty)</td>
<td>1. Person-male (previous)</td>
</tr>
<tr>
<td>2. Location-city</td>
<td>2. “car” (previous)</td>
</tr>
<tr>
<td>3. (empty) (previous)</td>
<td>3. Person</td>
</tr>
<tr>
<td>4. “game” (previous)</td>
<td>4. Person-male</td>
</tr>
<tr>
<td>5. “game”</td>
<td>5. “jury”</td>
</tr>
<tr>
<td>6. Person-male</td>
<td>6. Person (next)</td>
</tr>
<tr>
<td>7. “point” (previous)</td>
<td>7. (empty) (next)</td>
</tr>
<tr>
<td>8. “game” (next)</td>
<td>8. “point”</td>
</tr>
<tr>
<td>10. “win”</td>
<td>10. “point” (next)</td>
</tr>
<tr>
<td>11. (empty) (next)</td>
<td>11. “change” (previous)</td>
</tr>
<tr>
<td>12. “basketball”</td>
<td>12. “research” (next)</td>
</tr>
<tr>
<td>13. “point”</td>
<td>13. “fiber” (previous)</td>
</tr>
<tr>
<td>14. “title” (previous)</td>
<td>14. “retirement” (next)</td>
</tr>
<tr>
<td>15. “win” (previous)</td>
<td>15. “look”</td>
</tr>
</tbody>
</table>
Choosing an optimal number of features
Classifiers

- Naïve Bayes
- Decision Trees
- Support Vector Machines
- Voted Perceptrons
- Language Model
- AdaBoosted Naïve Bayes & Decision Stumps
- Maximum Entropy
Naïve Bayes

- Build a binary classifier (present/absent) for each concept.

\[
p(c \mid d_w) = \frac{p(d_w \mid c)p(c)}{p(d_w)}
\]
Language Modeling

- Conceptually similar to Naïve Bayes but
  - Multinomial
  - Smoothed distributions
  - Different feature selection
Maximum Entropy Classification

- Binary constraints
- Single 75-concept model
- Ranked list of concepts for each shot.
Results on the most common concepts

<table>
<thead>
<tr>
<th>Concept</th>
<th>Chance</th>
<th>Lang Model</th>
<th>Naïve Bayes</th>
<th>MaxEnt</th>
</tr>
</thead>
<tbody>
<tr>
<td>text</td>
<td>0.60</td>
<td>0.58</td>
<td>0.57</td>
<td>0.56</td>
</tr>
<tr>
<td>non_studio</td>
<td>0.55</td>
<td>0.52</td>
<td>0.51</td>
<td>0.50</td>
</tr>
<tr>
<td>face</td>
<td>0.50</td>
<td>0.49</td>
<td>0.48</td>
<td>0.47</td>
</tr>
<tr>
<td>indoors</td>
<td>0.45</td>
<td>0.43</td>
<td>0.42</td>
<td>0.41</td>
</tr>
<tr>
<td>outdoors</td>
<td>0.40</td>
<td>0.38</td>
<td>0.37</td>
<td>0.36</td>
</tr>
<tr>
<td>people</td>
<td>0.35</td>
<td>0.33</td>
<td>0.32</td>
<td>0.31</td>
</tr>
<tr>
<td>person</td>
<td>0.30</td>
<td>0.28</td>
<td>0.27</td>
<td>0.26</td>
</tr>
</tbody>
</table>
Results on selected concepts

- Chance
- Lang Model
- Naïve Bayes
- MaxEnt

Bar chart showing the performance of different models for various concepts such as weather, basketball, face, sky, indoors, beach, vehicle, and car.
Mean Average Precision

![Bar chart showing Mean Average Precision for different models: Chance, Language Model, SVM, Naive Bayes, Max Ent. The values range from 0 to 0.14, with the Language Model having the highest AP at approximately 0.13, followed by SVM, Naive Bayes, and Chance.]
Will this help for retrieval?

- “Find shots of a person diving into some water.”

- “Find shots of the front of the White House in the daytime with the fountain running.”

- “Find shots of Congressman Mark Souder.”
Will this help for retrieval?

- “Find shots of a person diving into some water.”
  - person, water_body, non-studio_setting, nature_non-vegetation, person_action, indoors

- “Find shots of the front of the White House in the daytime with the fountain running.”
  - building, outdoors, sky, water_body, cityscape, house, nature_vegetation

- “Find shots of Congressman Mark Souder.”
  - person, face, indoors, briefing_room_setting, text_overlay
### Performance on retrieval-relevant concepts

<table>
<thead>
<tr>
<th>Concept</th>
<th>Importance</th>
<th>AP</th>
<th>Chance</th>
</tr>
</thead>
<tbody>
<tr>
<td>outdoors</td>
<td>0.68</td>
<td>0.434</td>
<td>0.270</td>
</tr>
<tr>
<td>person</td>
<td>0.48</td>
<td>0.267</td>
<td>0.227</td>
</tr>
<tr>
<td>sky</td>
<td>0.40</td>
<td>0.119</td>
<td>0.061</td>
</tr>
<tr>
<td>vehicle</td>
<td>0.36</td>
<td>0.106</td>
<td>0.043</td>
</tr>
<tr>
<td>face</td>
<td>0.28</td>
<td>0.582</td>
<td>0.414</td>
</tr>
<tr>
<td>man-made-obj.</td>
<td>0.28</td>
<td>0.190</td>
<td>0.156</td>
</tr>
<tr>
<td>building</td>
<td>0.24</td>
<td>0.078</td>
<td>0.042</td>
</tr>
<tr>
<td>road</td>
<td>0.24</td>
<td>0.055</td>
<td>0.037</td>
</tr>
<tr>
<td>transportation</td>
<td>0.24</td>
<td>0.151</td>
<td>0.065</td>
</tr>
<tr>
<td>indoors</td>
<td>0.24</td>
<td>0.459</td>
<td>0.317</td>
</tr>
</tbody>
</table>
Summary

- Predict visual concepts for ASR
- Tried Naïve Bayes, SVMs, MaxEnt, Language Models,…
- Expect improvements in retrieval
Joint Visual-Text Video OCR

Proposed by:
Matthew Krause
Georgetown University
Motivation

- TREC queries ask for:
  - specific persons
  - specific places
  - specific events
  - specific locations
Motivation

- “Find shots of Congressman Mark Souder”
Motivation

“Find shots of a graphic of Dow Jones Industrial Average showing a rise for one day. The number of points risen that day must be visible.”
Motivation

- Find shots of the Tomb of the Unknown Soldier in Arlington National Cemetery.
Motivation
Joint Visual-Text Video OCR

- Goal: Improve video OCR accuracy by exploiting other information in the audio and video streams during recognition.
Why use video OCR?

- Sources tell C.N.N. there’s evidence that links those incidents with the January bombing of a women’s health clinic in Birmingham, Alabama. Pierre Thomas joins us now from Washington. He has more on the story in this live report...
Why use video OCR?
Why use video OCR?
Why use video OCR?

Those links are growing more intensive investigative focus toward fugitive Eric Rudolph who's been charged in the Birmingham bombing which killed an off-duty policeman...
Why use video OCR?

- Text overlays provide high precision information about query-relevant concepts in the current image.
Finding Text

- Use existing tools and data from IBM/CMU.
Image Processing

- Preprocessing
  - Normalize the text region’s height
- Feature extraction
  - Color
  - Edge Strength and Orientation
Proposal: HMM-based recognizer
Proposal: Cache-based LMs

- Augment the recognizers with an interpolation of language models
  - Background language model
  - Cache-based language model
  - ASR or closed caption text
  - “Interesting” words from the cache
  - Named Entities

\[
p(c_i \mid h) = p_{bg}(c_i \mid h)^{\lambda_1} p_{cache}(c_i \mid h)^{\lambda_2} p_{interest}(c_i \mid h)^{\lambda_3}
\]
Evaluation

- Evaluate on TRECVID data
- Character Error Rate
  - Compare vs. manual transcriptions
- Mean Average Precision
  - NIST-provided relevance judgments
Summary

- Information from text overlays appears to be useful for IR.
- General character recognition is a Hard problem.
- Adding in external knowledge sources via the LMs should improve accuracy.
Work Plan

1. **Text Localization**
   - IBM/CMU text finders + height normalization

2. **Image Processing & Feature Extraction**
   - Begin with color and edge features

3. **HMM-based Recognizer**
   - Train using TREC data with hand-labeled captions

4. **Language Modeling**
   - Background, Cache, and “Interesting Words”
Retrieval Experiments and Summary

Presented by Dietrich Klakow
Presentation Outline

Translation (MT) models (Paola),

Relevance Models (Shao Lei, Desislava),

Graphical Models (Pavel, Brock)

Text classification models (Matt)

Integration & Summary (Dietrich)
The Matrix

Document

Words $d_w$  Visterms $d_v$

Query

Visterms $q_v$  Words $q_w$

$$p(q_w, q_v \mid d_w, d_v)$$
The Matrix

Document

Words $d_w$  Visterms $d_v$

$p(q_w | d_w)$  $p(q_w | d_v)$

$p(q_v | d_w)$  $p(q_v | d_v)$

Query

Visterms $q_v$  Words $q_w$
The Matrix

Document

Query

Visterms $q_v$ Words $d_w$

\[ p(q_v | d_w) \]
- Naïve Bayes
- Max. Ent
- LM
- SVM, Ada Boost, …

Visterms $d_v$

\[ p(q_w | d_v) \]
- MT
- Relevance Models
- HMM

\[ p(q_v | d_v) \]
Retrieval Model I: $p(q \mid d)$

\[ p(q_w, q_v \mid d_w, d_v) \]

\[ = p(q_w \mid d_w, d_v) \times p(q_v \mid d_w, d_v) \]

\[ p(q_w \mid d_w, d_v) \]

\[ = \lambda_w p(q_w \mid d_w) + (1 - \lambda_w) p(q_w \mid d_v) \]
Retrieval Model I: $p(q|d)$

\[
p(q_w, q_v | d_w, d_v) = \\
\left[ \lambda_w p(q_w | d_w) + (1 - \lambda_w) p(q_w | d_v) \right] \times \\
\left[ \lambda_v p(q_v | d_w) + (1 - \lambda_v) p(q_v | d_v) \right]
\]

$\alpha$ Only minor improvements over baseline
Retrieval Model II: \( p(q \mid d) \)

- We want to estimate \( p(q, q_v, d_w, d_v) \)
- Assume pairwise marginals given:

\[
\sum_{q_v, d_w} p(q_w, q_v, d_w, d_v) = p(q_w, d_v)
\]

- Setting: Maximum Entropy problem
  - 4 constraints
  - 1 iteration of GIS:

\[
p(q_w, q_v \mid d_w, d_v) \propto p(q_w \mid d_w)^{\lambda_1} p(q_w \mid d_v)^{\lambda_2} p(q_v \mid d_w)^{\lambda_3} p(q_v \mid d_v)^{\lambda_4}
\]
Baseline TRECVID: Text Retrieval

Retrieval mAP: 0.131
Combination with visual model

\[ p(q_w | d_w) \quad p(q_w | d_v) \]

Document

Query

Words \( d_w \)

Visterms \( d_v \)

mAP: 0.131
Combination with visual model

Document

Words $d_w$  Visterms $d_v$

$p(q_w | d_w)$  $p(q_w | d_v)$

Query

Visterms $q_v$  Words $q_w$

MT: Best overall performance so far

Retrieval mAP: 0.139

Concept Annotation on images
mAP on TRECVID

<table>
<thead>
<tr>
<th>Model</th>
<th>mAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>MT</td>
<td>0.126</td>
</tr>
<tr>
<td>Relevance Models</td>
<td>0.158</td>
</tr>
<tr>
<td>HMM</td>
<td>0.145</td>
</tr>
</tbody>
</table>

MT: Best overall performance so far
Combination with MT and ASR

Concept Annotation on images:
mAP on TRECVID

<table>
<thead>
<tr>
<th></th>
<th>MT</th>
<th>Relevance Models</th>
<th>HMM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Retrieval mAP</td>
<td>0.149</td>
<td>0.158</td>
<td>0.145</td>
</tr>
<tr>
<td>Best results reported in literature: retrieval mAP</td>
<td>0.162</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Recall-Precision-Curve

Improvements in high precision region
Difficulties and Limitations we faced

- Annotations are Inconsistent, sometimes abstract, ...
- Used plain vanilla features
  - Color, texture, edge on key-frames
  - No time for exploration of alternatives
- Uniform block segmentation of images
- Upper bound for concepts from ASR
Future Work

- **Model**
  - Incompletely labelled images
  - Inconsistent annotations

- **Get beyond the 75-concept bottleneck**
  - Larger concept set (+training data)
  - Direct modelling

- **Better model for spatial and temporal dependencies in video**

- **Query dependent processing**
  - E.g. image features, combination weights, OCR-features

Desislava

Shaolei and Brock

Matt
Overall Summary

- **Concepts from image**
  - MT: CLIR with direct translation works best
  - Relevance models: best numbers on development test
  - HMM: novel competitive approach for image annotation

- **Concepts from ASR:**
  - oh my god, it works

- **Fusion:**
  - adding multiple source in log-linear combination helped

- **Overall:** 14% improvement
Acknowledgments

- TREC for the data
- BBN for NE-tagging
- IBM:
  - for providing the features
  - Close captioning alignment (Arnon Amir)
- Help with GMTK: Jeff Bilmes and Karen Livescu
- CLSP for the capitalizer (WS 03 MT-team)
- INRIA for the face detector
- NSF, DARPA and NSA for the money
- CLSP for hosting
  - Laura, Sue, Chris
  - Eiwe, John, Peter
  - Fred