Towards Content Based Image Retrieval on AIA Images

by
Dustin Kempton
October 2, 2019
Agenda

• Motivation
• Sparse Coding
• Input Data
• Whole Image Descriptors and Retrieval
• Image Region Descriptors and Retrieval
Motivation
The SDO Dataset

• SDO produces over 70,000 images per day - 1.5 Terabytes daily of solar images

• 100 Millionth AIA image on Jan. 19, 2015

• Equates to almost 5 Petabytes so far
SDO Dataset is Large

Size of symbol represents nominal data rate

Nominal Lifetime Data Volume (Petabyte)

100,000
10,000
1,000
100
10
1

Start Year

Courtesy Joseph Gurman NASA Goddard Space Flight Center
SDO’s Growing Dataset

• At with more than 100 Million images from the AIA, how do we find what we want?
• We need an efficient mechanism for searching the image data.
Image Retrieval by Annotation

• Traditional text-based image search engines
  – Images are annotated manually
  – Use text-based retrieval methods
Limitations of Annotations

• Problem of image annotation
  – Large volumes of databases – no single person can annotate that many images
  – Valid for only one language – retrieval of an image should be universal

• Problems of using humans
  – Human perception can sometimes be subjective (who remembers the viral articles about is the dress white or blue?)
  – Humans have biases and interject those into their labels
  – Puts too much responsibility on the End User to get the query right
Using Semi-Automated Annotation

• But wait, we have labeled regions!
  – What if it’s a new phenomena?
  – How about rare phenomena?
• Additionally, annotation doesn’t support abstract concepts
  – Queries that are difficult to described in words, but is simply some visual feature of the image?
Image Retrieval

- The images themselves have rich content that can be used for retrieval!
- The most common task is to answer: How can I find images that meet some description or are like an example?
- This is called “content-based image retrieval” (CBIR).
  - Image content is used, rather than metadata.

- Using example images:
  - Takes the responsibility of forming the query away from the user.
  - Allows each image to be described by its own features.
Image Retrieval on SDO’s Growing Dataset

• Whole image retrieval was previously explored, but selectivity was an issue
• Indexing was not used due to dimensionality of descriptors
  – Lookup tables were computed
  – Does not work for continuously growing dataset
• An effective region describing method was also lacking
The CBIR Problem

• Need to produce image descriptors for solar images
  – With constraint on dimensionality so they can be used with existing indexing techniques
  – We developed a sparse coding image descriptor method
The sparse coding descriptor looks to address both the selectivity and dimensionality issue.

- We show improvement on the selectivity issue.
- While keeping the descriptor vector constrained to a size that current indexing algorithms can handle.
Sparse Coding

By Learn a dictionary of basis vectors
Sparse Coding Defined

• Unsupervised learning technique
  – For learning basis dictionary \( D \in \mathbb{R}^{m \times k} \)
  – Linear model \( x \approx D\alpha \) or
    \[ x = D\alpha + \epsilon \]
  • \( \alpha \) has few significantly non-zero coefficients
Our objective is to create a dictionary for sparse representation of a finite set of signals $X = [x_1, ..., x_n]$ with $x \in \mathbb{R}^m$

Equates to the optimization of the cost function

$$f_n(D) = \frac{1}{n} \sum_{i=1}^{n} \ell(x_i, D)$$

With $D$ in $\mathbb{R}^{m \times k}$ being the dictionary

$\ell$ is a loss function such that $\ell(x, D)$ is small if $D$ is “good”
Once the dictionary is learned each of the signals in $X$ can be described as a linear combination of elements in $D$:

$$x_i = D\alpha_i + \epsilon$$

again using

$$\min_{\alpha \in \mathbb{R}^k} \frac{1}{2} \|x - D\alpha\|_2^2 + \lambda \|\alpha\|_1$$
Sparse Data Representation

• Vector of weights are sparse
  – Allows for compression
Sparse Data Representation
Sparse Coding Illustration

Input Region

Learned Bases \((d_1, d_2, \ldots, d_n)\)

\[\approx 0.8 \ast + 0.3 \ast + \cdots\]
Image Reconstruction

• The image can be reconstructed from the sparse representation of the original signal.
• As per the sparse constraint most of the coefficients are a zero value.
• Larger dictionaries can do better at recreation
  – We just want to describe the image for indexing, so smaller is better
Image Reconstruction Example

- Start with a 171Å image
- Example using software library developed at GSU DM Lab
  - DMLabLib Project
  - DMLabLib V-0.0.4-SNAPSHOT Documentation

Original Image
Image Reconstruction Example

Original Image

Mean Patch Values
Image Reconstruction Example

Sparse Reconstruction Values

Reconstructed Image
Image Reconstruction Example

Original Image

Reconstructed Image
Image Reconstruction Stats

• Patch size was 8 by 8 pixel
• Dictionary size was 256
  – Only learned on/for one image
• Sparse vector was limited to about 20 non-zero values
  – These weight vectors are our descriptors for the 8 by 8-pixel regions
Input Data
Atmospheric Imaging Assembly (AIA)

• SDO’s AIA instrument produces high resolution images
  – 4096 by 4096 pixels
  – 9 wavelengths (that we use)
  – Image about every 12 seconds
  – 700 GB daily

Credit: NASA
Prior CBIR work designed image parameter extraction process.

- We recently improved and expanded the date range of this dataset

Each wavelength has 10 parameters calculated on them

<table>
<thead>
<tr>
<th>Label</th>
<th>Name</th>
<th>Equation</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>Entropy</td>
<td>[ E = -\sum_{i=0}^{L-1} p(x_i) \log_2 p(x_i) ]</td>
</tr>
<tr>
<td>P2</td>
<td>Mean</td>
<td>[ m = \frac{1}{L} \sum_{i=0}^{L-1} x_i ]</td>
</tr>
<tr>
<td>P3</td>
<td>Std. Deviation</td>
<td>[ \sigma = \sqrt{\frac{1}{L} \sum_{i=0}^{L-1} (x_i - m)^2} ]</td>
</tr>
<tr>
<td>P4</td>
<td>Fractal Dim.</td>
<td>[ D_0 = -\lim_{\varepsilon \to 0} \left( \frac{\log N(\varepsilon)}{\log \varepsilon} \right) ]</td>
</tr>
<tr>
<td>P5</td>
<td>Skewness</td>
<td>[ \mu_3 = \sum_{i=0}^{L-1} (x_i - m)^3 p(x_i) ]</td>
</tr>
<tr>
<td>P6</td>
<td>Kurtosis</td>
<td>[ \mu_4 = \sum_{i=0}^{L-1} (x_i - m)^4 p(x_i) ]</td>
</tr>
<tr>
<td>P7</td>
<td>Uniformity</td>
<td>[ U = \sum_{i=0}^{L-1} p^2(x_i) ]</td>
</tr>
<tr>
<td>P8</td>
<td>Rel. Smoothness</td>
<td>[ R = 1 - \frac{1}{1 + \sigma^2(x)} ]</td>
</tr>
<tr>
<td>P9</td>
<td>T. Contrast</td>
<td>See Tamura et al. 1978</td>
</tr>
<tr>
<td>P10</td>
<td>T. Directionality</td>
<td>See Tamura et al. 1978</td>
</tr>
</tbody>
</table>
Image Parameters

• Images are broken up into a grid of cells
  – Each cell is 64 pixel by 64 pixels in size

A heatmap of Tamura directionality over the entire segmented image

An SDO image in the original size

A single cell
Image Parameters

- Each image becomes a cube of parameter values
  - The cube is our starting dataset
Whole Image Descriptors
Sparse Image Descriptors

- Whole image description using max-pooled sparse vectors
  - Used sparse vectors to describe overlapping regions of images
  - Pooled each coefficient vector using max-pooling
Image Parameters for Description

- Start with the cube of parameters described earlier
Creating the Input Signal of an Image

- The data cube of image parameters is the raw data.
- We extract multiple windows of size $n$ by $n$ to create the input signals.
Creating the Input Signal of an Image

- The area of interest is represented by a matrix of cell values
- Which are collimated to produce vector $x_i \in X$
Dictionary per Wavelength

• A dictionary was learned for each wavelength of images
  – The learning is on a large sample of windows selected from numerous images

• Example dictionary learned on actual pixel values
Pooling Sparse Descriptors

- Each of the signal vector sparse representations ($\alpha_i \in A$) are max-pooled on coefficients
- Creates one vector per image

<table>
<thead>
<tr>
<th>$\alpha_{11}$</th>
<th>$\alpha_{21}$</th>
<th>$\alpha_{m1}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\alpha_{12}$</td>
<td>$\alpha_{22}$</td>
<td>$\alpha_{m2}$</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>$\alpha_{1n-1}$</td>
<td>$\alpha_{2n-1}$</td>
<td>$\alpha_{m_{n-1}}$</td>
</tr>
<tr>
<td>$\alpha_{1n}$</td>
<td>$\alpha_{2n}$</td>
<td>$\alpha_{m_n}$</td>
</tr>
</tbody>
</table>

$max_i |\alpha_{i1}|$
Sparse Descriptors

<table>
<thead>
<tr>
<th>Original Image</th>
<th>Parameter Cube</th>
<th>Sparse Descriptor</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Image" /></td>
<td><img src="cube1.png" alt="Cube" /></td>
<td><img src="descriptor1.png" alt="Descriptor" /></td>
</tr>
<tr>
<td><img src="image2.png" alt="Image" /></td>
<td><img src="cube2.png" alt="Cube" /></td>
<td><img src="descriptor2.png" alt="Descriptor" /></td>
</tr>
<tr>
<td><img src="image3.png" alt="Image" /></td>
<td><img src="cube3.png" alt="Cube" /></td>
<td><img src="descriptor3.png" alt="Descriptor" /></td>
</tr>
</tbody>
</table>

- Repeated for all 9 AIA wavelengths in our dataset
Distance Measures

- We considered Minkowski and fractional $p$ norms.

\[ \|x_j\|_p = \left( \sum_i |x_{ij}|^p \right)^{\frac{1}{p}} \]

- For an $x_j = [x_{j1}, \ldots, x_{jd}] \in \mathbb{R}^d$
- Minkowski is $1 \leq p \leq \infty$
- Fractional is $0 < p < 1$
Measures on Parameters

- Evaluated by summing the temporal distance of the first nearest neighbor in each distance measure
  - Started with original parameter space

\[
\begin{align*}
\sum_{i=1}^{1000} |t_q - t_r| 
\end{align*}
\]

- \( t_q \) = time of query
- \( t_r \) = time of first nearest neighbor query result

Sum if result is always the most temporally nearest neighbor
Distance Distribution on Parameters

- Evaluated the distribution of temporal distance for the first nearest neighbor
  - Original parameter space

Result is almost always the most temporally close image in the database.

Not an interesting result
Measures on Pooled Sparse Vectors

- Evaluated by summing the temporal distance of the first nearest neighbor in each distance measure
  - Then pooled descriptors
Distance Distribution
Pooled Sparse Vectors

- Evaluated the distribution of temporal distance for the first nearest neighbor
  - Pooled descriptors
Distances Over Time

Similarity of Image Descriptors Jan 20-23, 2012

Pooled  Mean Param  All Params

Less  More

Similarity
Nearest Neighbor Query Example

- Pair chosen because they are the nearest neighbor query results with the greatest temporal distance between the query and its first nearest neighbor.
Most Distant Neighbor Query Example

- Pair chosen because they are the most distant neighbor query results with the greatest temporal distance between the query and its maximal distance neighbor (Most dissimilar image).
Image Region Descriptors
For a baseline we start with statistical description of the parameter values in a region
- Use a standard 7 statistic set as a set of features for each parameter
  - Minimum parameter value
  - First Quartile
  - Second Quartile (Median)
  - Third Quartile
  - Maximum parameter value
  - Average parameter value
  - Standard Deviation of parameter values
Statistical Descriptor Ranking

• Use the F-Statistic ranking procedure
  – Using three classes, Active Region, Coronal Hole, and Quiet Sun
  – Performed on the Statistical Features not the parameter values
Sparse Vector ROI

- Same extraction method from tracking is used
- Extracting windows from the ROI for each event type to create $X$
Pooling Sparse Region Descriptors

• Each of the signal vector sparse representations ($\alpha_i \in A$) are sum-pooled on coefficients

• Creates one vector per event

\[
\begin{align*}
\sum \alpha_{ij} & \quad \text{if } \alpha_{ij} \geq 0 \\
\sum |\alpha_{ij}| & \quad \text{if } \alpha_{ij} < 0
\end{align*}
\]
Region Classification Experiments

Have two datasets
- 3-Class using AR, CH, QS
- 5-Class using AR, CH, QS, SS, SG

Split event reports into months they occur

Use 2/3 months training 1/3 months testing
- Use all combinations of 2/3 and 1/3
- Random under sampling to balance classes
- 10 repetitions of under sampling
Stat Feature
Results 3-Class

- Best Results around 12 Features
  - AR Median 89%
  - CH Median 78.6%
  - QS Median 39.2%
  - Combined Media 71.1%
A dictionary is learned using the input regions.

We tried various dictionary sizes and number of parameters. Parameters were ranked similar to tracking by using histogram intersection.

The sparse pooled vector is the input into the classifier.
Sparse Vector Results 3-Class

- Best Results using all 90 image parameters
  - AR Median 93.1%
  - CH Median 69.7%
  - QS Median 77.4%
  - Combined Media 80.2%
- Parameters compressed to 40 dimensions by using dictionary of that size
5-Class Dataset Stats

Report Counts
- AR = 13,518
- CH = 10,780
- QS = 13,518
- SG = 7,805
- SS = 3,417

Percentage of Dataset
- AR = 27.57%
- CH = 21.98%
- QS = 27.57%
- SG = 15.92%
- SS = 6.97%
5-Class Classification Results AR

Stat Feature Vector

AR: 39.7%
SG: 17.3%
SS: 30.5%
CH: 13.0%
QS: 11.4%

Sparse Feature Vector

AR: 51.3%
SG: 20.1%
SS: 21.5%
CH: 3.2%
QS: 9.9%
5-Class Classification Results CH

Stat Feature Vector

- CH: 40.7%
- QS: 53.3%

Sparse Feature Vector

- CH: 79.8%
- QS: 17.1%
5-Class Classification Results QS

Stat Feature Vector

Sparse Feature Vector

Kempton
5-Class Classification Results SS

Stat Feature Vector

Sparse Feature Vector
5-Class Classification Results SG

Stat Feature Vector

- SG: 17.1%
- CH: 24.8%
- QS: 35.2%
- AR: 21.5%

Sparse Feature Vector

- SG: 28.5%
- CH: 11.7%
- QS: 27.8%
- AR: 4.2%
5-Class KNN Retrieval AR

- Percentage of neighbors from each class for 100 random queries
  - As $K$ varies from 1 to 200
5-Class KNN Retrieval CH

- Percentage of neighbors from each class for 100 random queries
  - As $K$ varies from 1 to 200
5-Class KNN Retrieval QS

- Percentage of neighbors from each class for 100 random queries
  - As $K$ varies from 1 to 200

![Graph showing percentage of neighbors for each class as $K$ varies from 1 to 200.](image)
5-Class KNN Retrieval

- Percentage of neighbors from each class for 100 random queries
  - As $K$ varies from 1 to 200

![Graph showing percentage of neighbors for each class](image)
SS Error or Not

- A number of Sun Spot events co-occur with the Active Region Reports
- SS being classified as AR, not always wrong
5-Class KNN Retrieval SG

- Percentage of neighbors from each class for 100 random queries
  - As $K$ varies from 1 to 200
Concluding Recap

• Showed the usefulness of sparse coding for image descriptors
  – Started with whole image descriptors
  – Utilized region descriptors similar to those developed in tracking
Future Work

• Develop system for both whole and region of images
  – Utilize indexing
    • Investigate multiple scales across multiple indexes
    • Investigate multi-layer pooling and coding as an alternative
  – Develop processing pipeline for images
    • Produce descriptors for images as received
Thank you for your attention.

Any Questions?
Dustin Kempton, Ph.D.
Department of Computer Science
Georgia State University
dkempton1@cs.gsu.edu

CONTACT INFORMATION
Publications From This Work


References


