Predictable Dual-View Hashing
Mohammad Rastegari, Jonghyun Choi, Shobeir Fakhraei, Hal Daumé III and Larry S. Davis
University of Maryland, College Park, MD, USA

Integrate different modalities
- Text – Image
- Sound – Text

Challenges
- Modalities are not directly comparable
- High dimensional modalities
- Efficient data structure for search

Previous approaches
- Domain specific [Farhadi et al. 2010]
- CCA-based [Gong et al. 2011, Sharma et al. 2012]

Dual-view hashing
Textual Space
- Human riding a horse
- Person laughing at restaurant
- Cow standing in a farm

Visual Space
- Horse
- Restaurant
- Farm

Binary code assignment
101 001
100 110
011 011
000 110

Optimization
\[
\min_{w_y} \| sgn(W^T_y X_Y) - sgn(W^T_f X_Y) \|_2^2
\]

Trivial solution: both \( w_y \) are zero

\[
\min_{w_y, w_f} \| B_T X_Y - B_f \|_2 + \| B_f B^T_f - I \|_2
\]

\[
+ \| W^T_f X_Y - B_Y \|_2 + \| B_Y B^T_f - I \|_2
\]

s.t.
\[
B_T = sgn(W^T_T X_Y),
B_Y = sgn(W^T_Y X_Y).
\]

Optimization is non-convex and combinatorial

\[
\min_{w_y, w_f} \| B_T B^T_f - I \|_2 + \| B_f B^T_f - I \|_2
\]

\[
+ \sum ||w_y|| + \sum ||w_f|| + C_1 \sum \xi_f + C_2 \sum \xi_f
\]

s.t.
\[
B_T = sgn(W^T_T X_Y),
B_Y = sgn(W^T_Y X_Y),
\]

\[
B^T_f (w_Y^* X_Y^*) \geq 1 - \xi_f \quad \forall i, j
\]

Using a block coordinate descent algorithm

How we do it
Step 1
Modality 1
0110
0101
1110
1010
0011
1101
1001

Step 1+1
Modality 2
0010
0101
1110
1010
0111
1101
1001

Predictability
Each bit should be predictable based on the neighbors

Qualitative results (w/ 32-bit codes)

Algorithm
Algorithm 1 Predictable Dual-View Hashing
Input: \( X_Y, X_f \in \mathbb{R}^{N \times d} \)
Output: \( B_T, B_Y \in \mathbb{R}^{N \times k} \)
1: \( W_y, W_f \in \mathbb{R}^{k \times k} \leftarrow CCA(X_Y, X_f, k) \)
2: \( B_Y = sgn(W^T_y X_Y) \)
3: \( B_Y = sgn(W^T_f X_f) \)
4: repeat
5: \( W_y \leftarrow \) Weights of k linear SVMs (for i\textsuperscript{th} SVM: training features are columns of \( X_Y \) and training labels are elements of \( i\textsuperscript{th} \) row of \( B_T \))
6: \( B_T = sgn(W^T_T X_Y) \)
7: Update \( B_Y \) using Eq. (5)
8: \( W_f \leftarrow \) Weights of k linear SVMs (for i\textsuperscript{th} SVM: training features are columns of \( X_f \) and training labels are elements of \( i\textsuperscript{th} \) row of \( B_Y \))
9: \( B_T = sgn(W^T_T X_f) \)
10: Update \( B_Y \) using Eq. (5)
11: until convergence
12: \( B_Y = sgn(W^T_Y X_Y) \)
13: \( B_T = sgn(W^T_T X_f) \)