Symposium on Social Multimedia and Cyber-Physical-Social Computing

Computing Visual Similarity with Social Context

Shuqiang Jiang

Institute of Computing Technology, Chinese Academy of Sciences

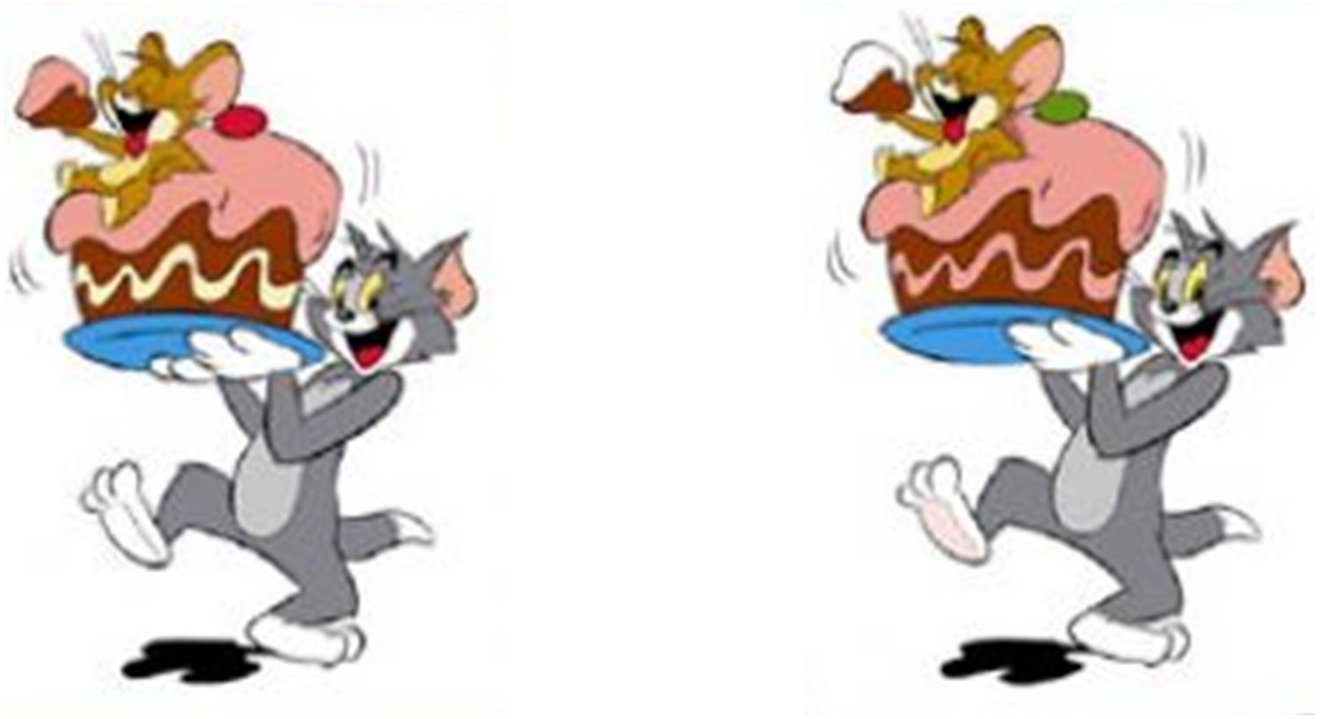
Aug. 15, 2013

中国科学院计算技术研究所
Institute of Computing Technology,Chinese Academy of Sciences

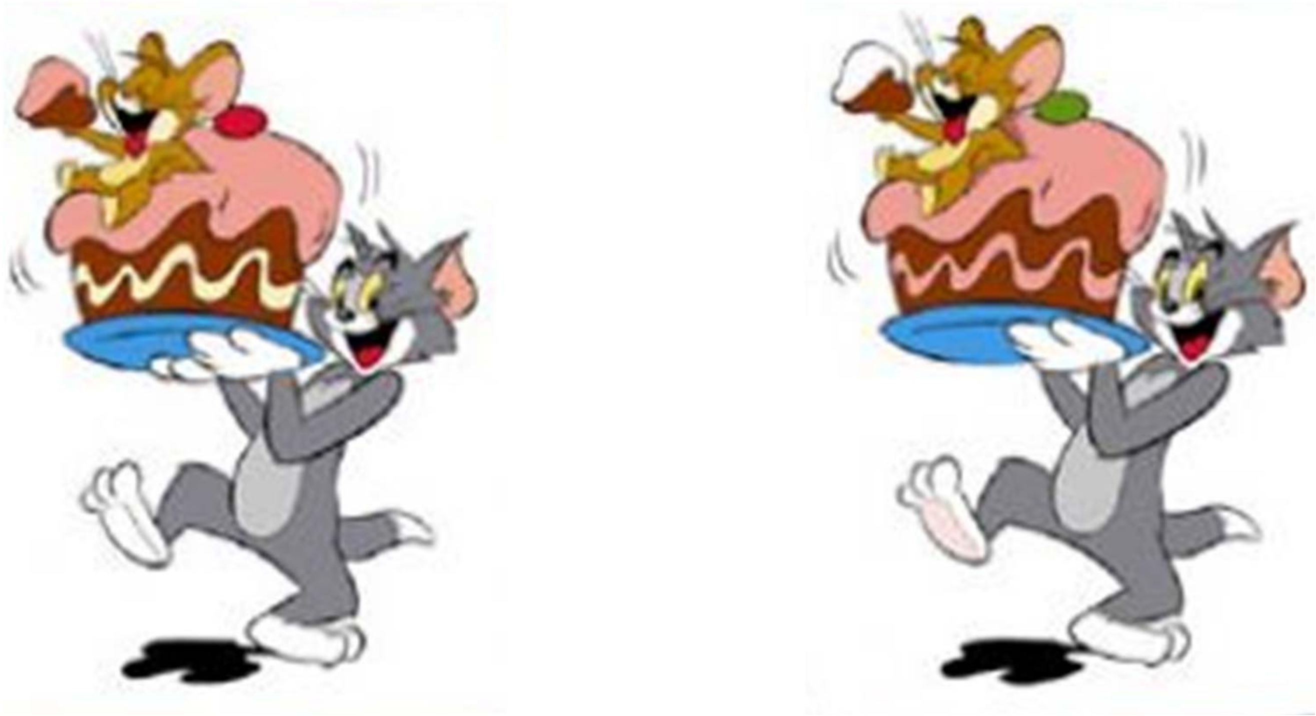
Find difference

中科院计算所

Institute of Computing Technology, Chinese Academy of Sciences



Find difference



four differences

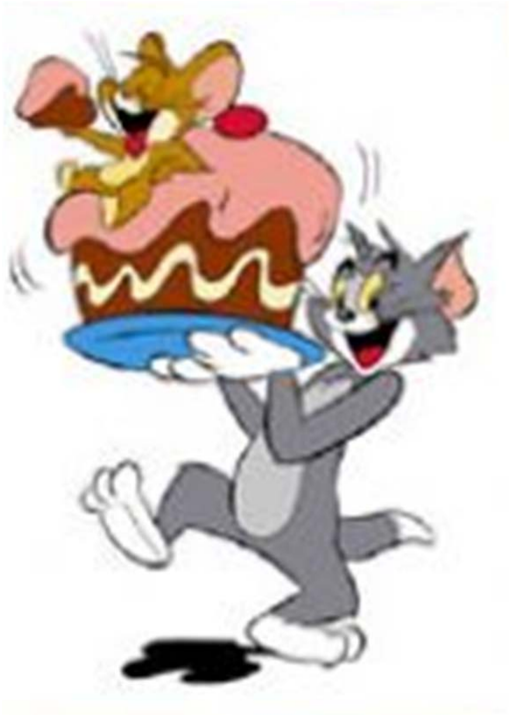
Find difference

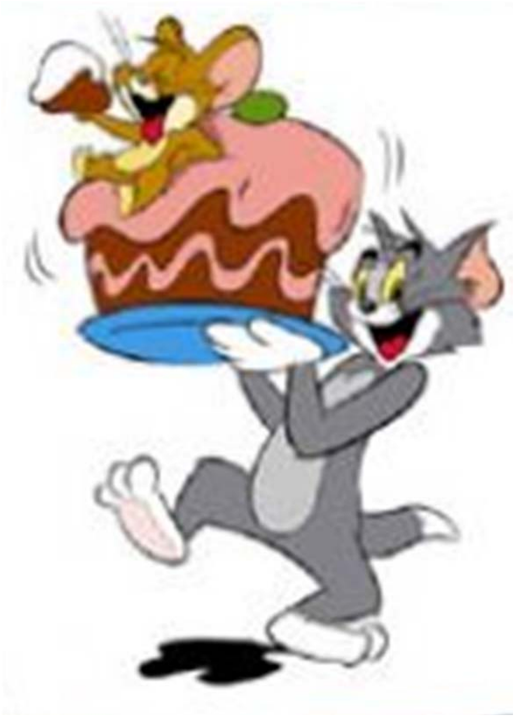
中科院计算所

Institute of Computing Technology, Chinese Academy of Sciences

four differences

Are they similar?

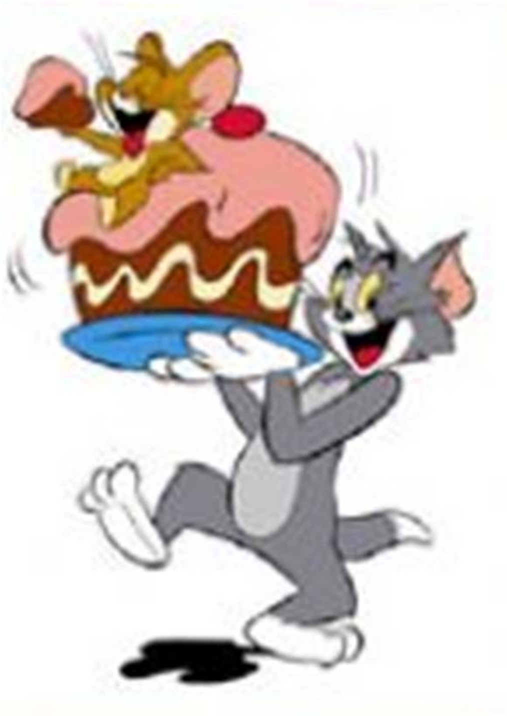


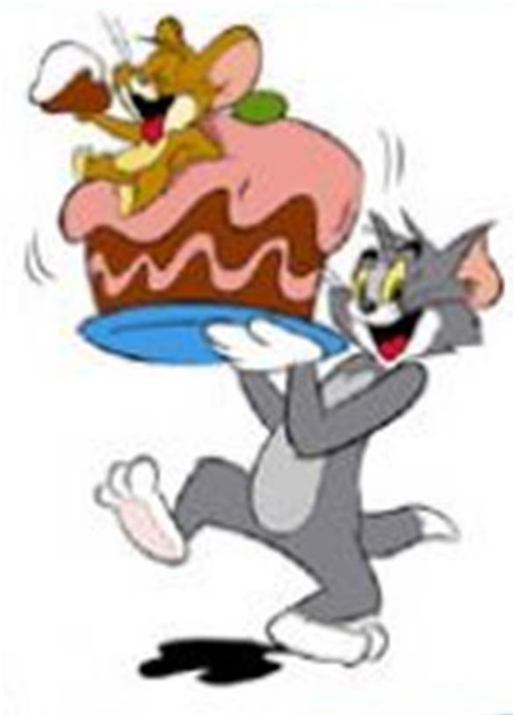


Are they similar?

中科院计算所

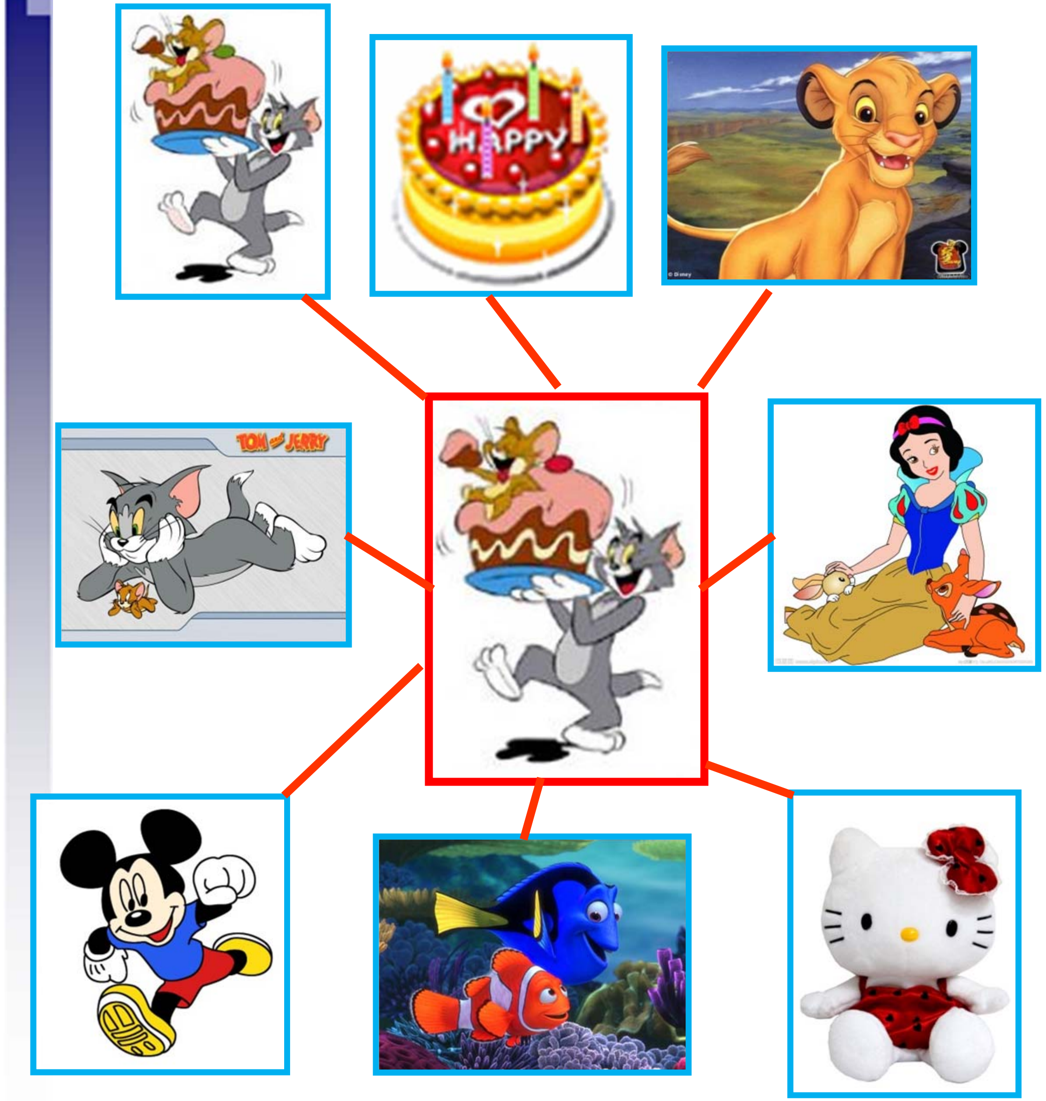
Institute of Computing Technology, Chinese Academy of Sciences



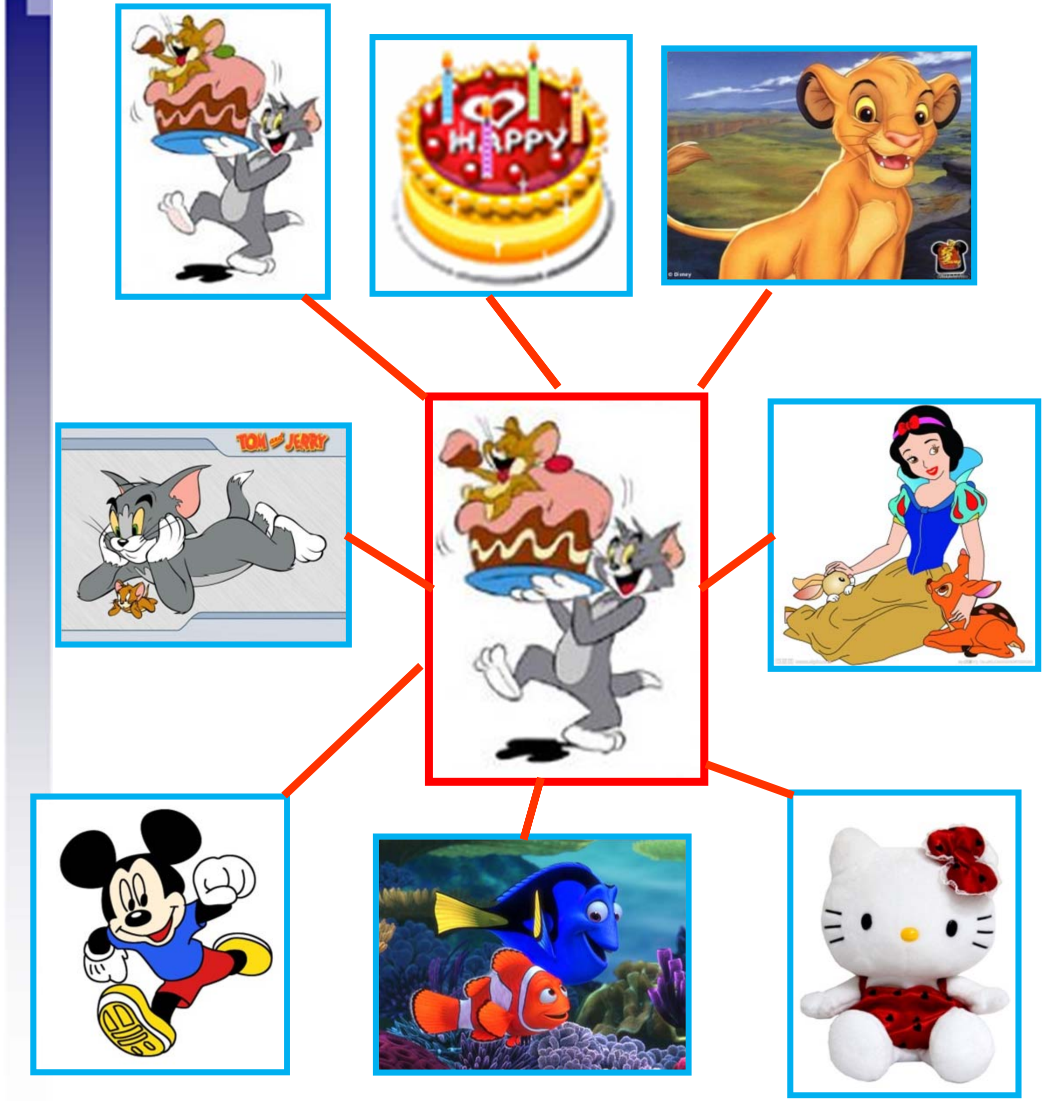


Near Duplicate

Multiple faces of image similarity



Multiple faces of image similarity



Same

Near Duplicate

Partial Duplicate

Visually Similar

Containing same object

Conceptually related

Contextually related

Multiple faces of image similarity

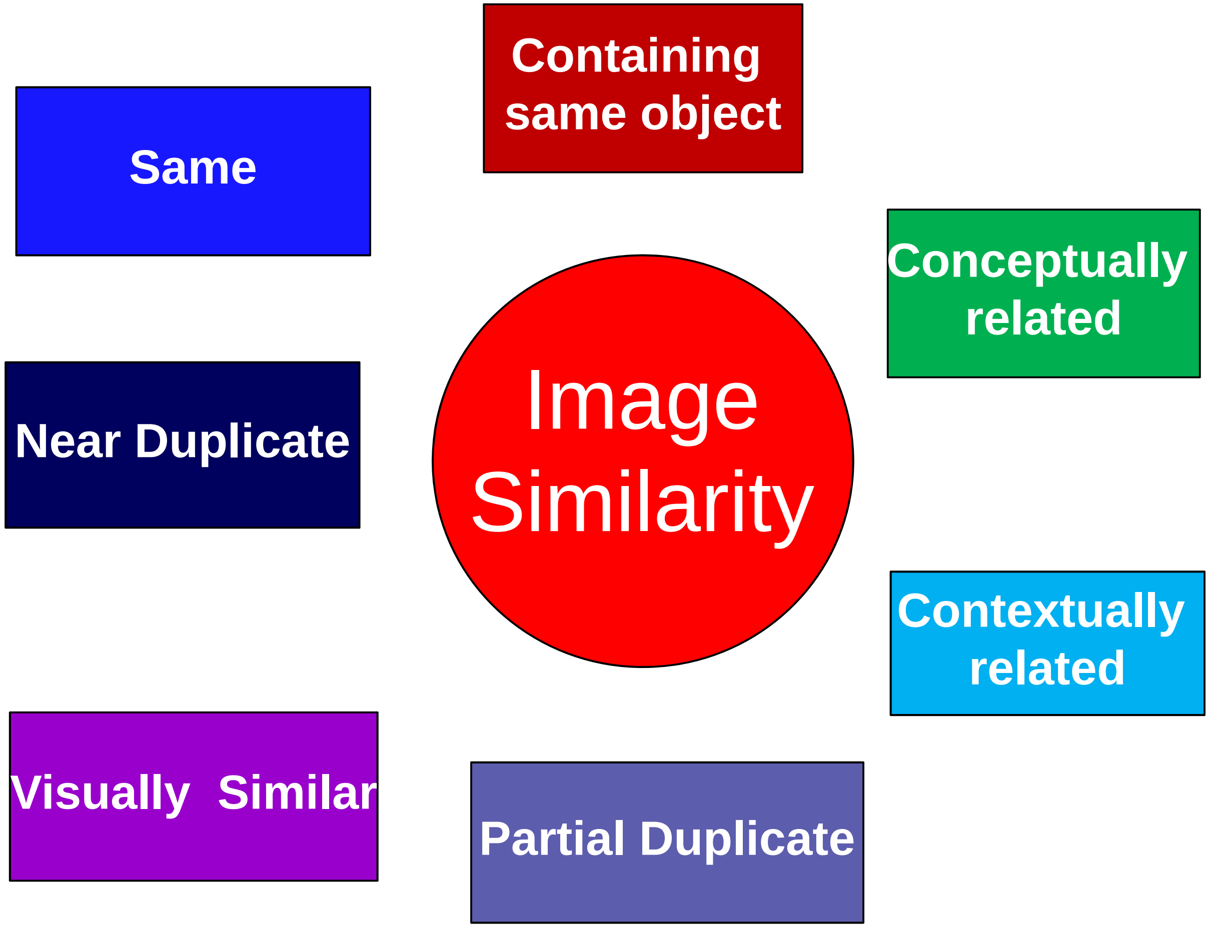


Image Similarity

Same

Containing same object

Conceptually related

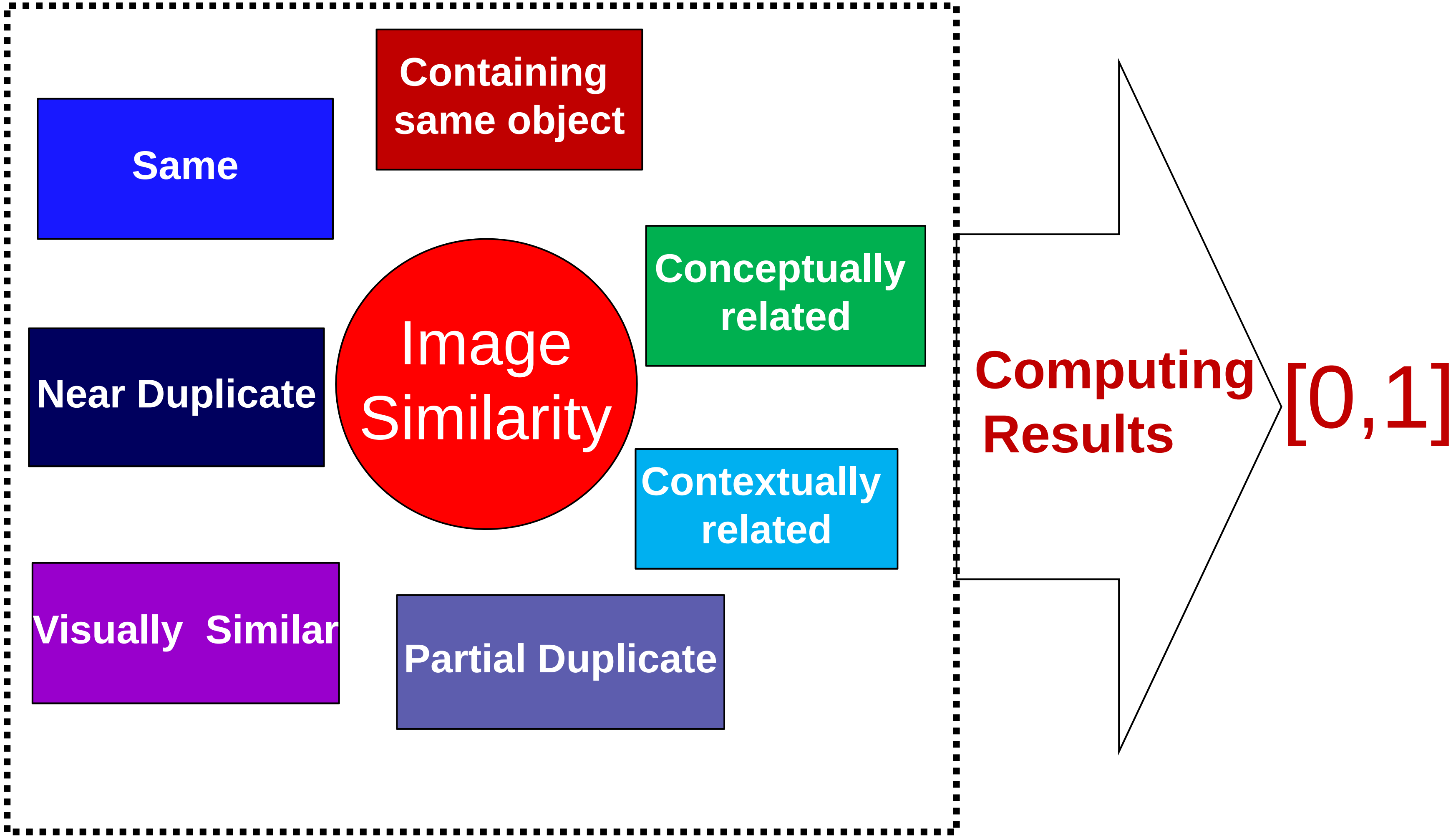
Near Duplicate

Contextually related

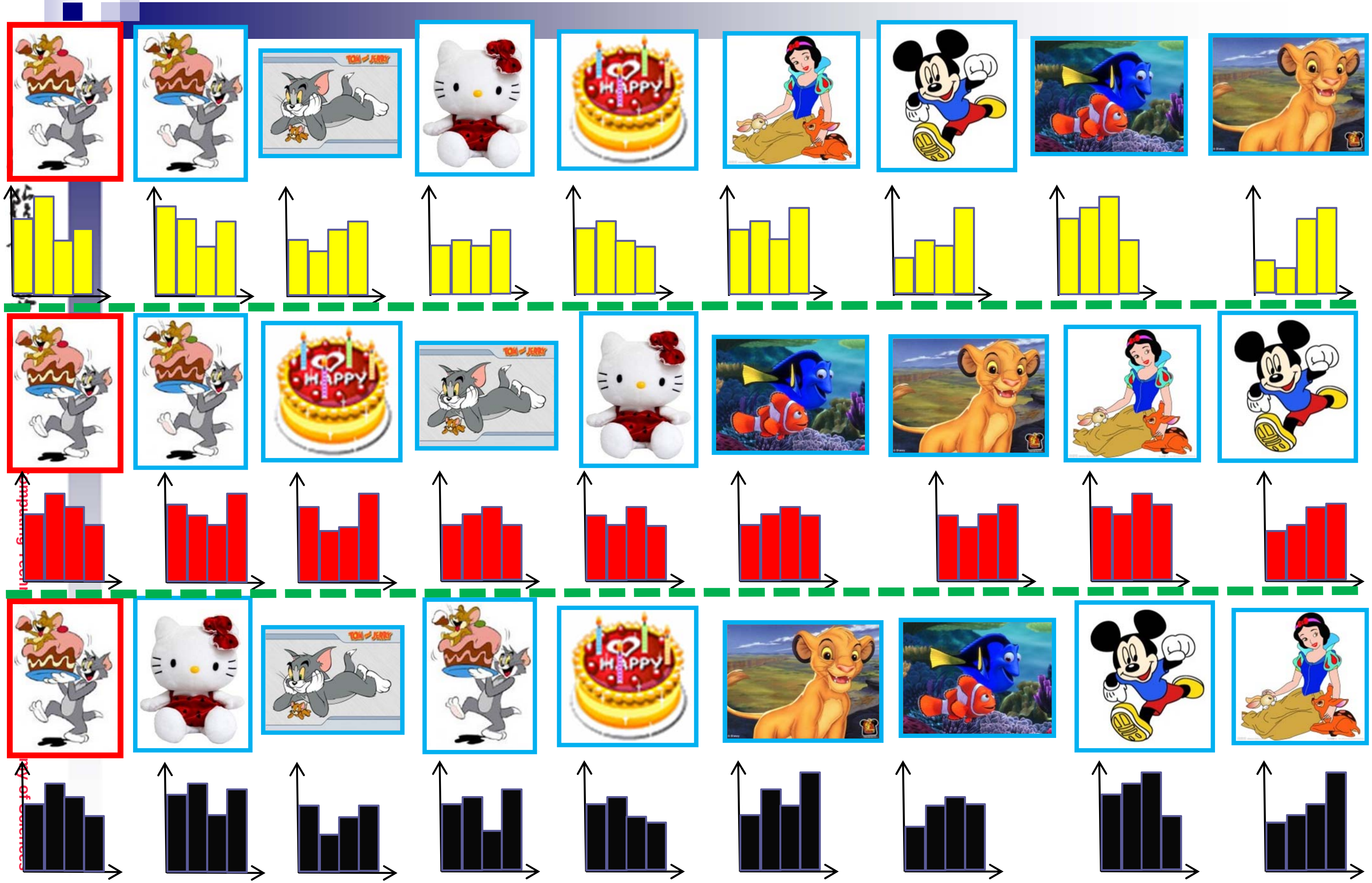
Visually Similar

Partial Duplicate

How to compute image similarity



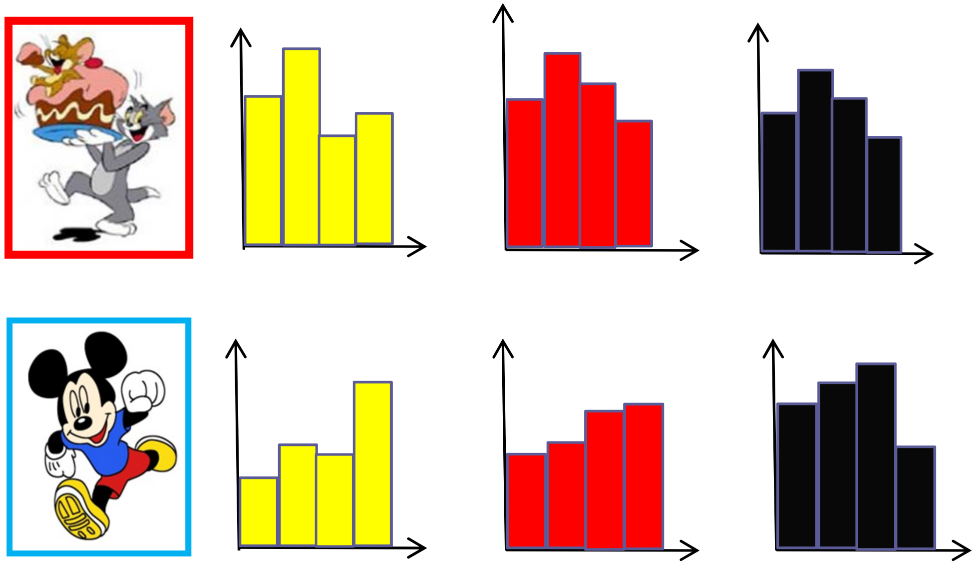
How to compute image similarity



How to compute image similarity

Traditional Solutions:

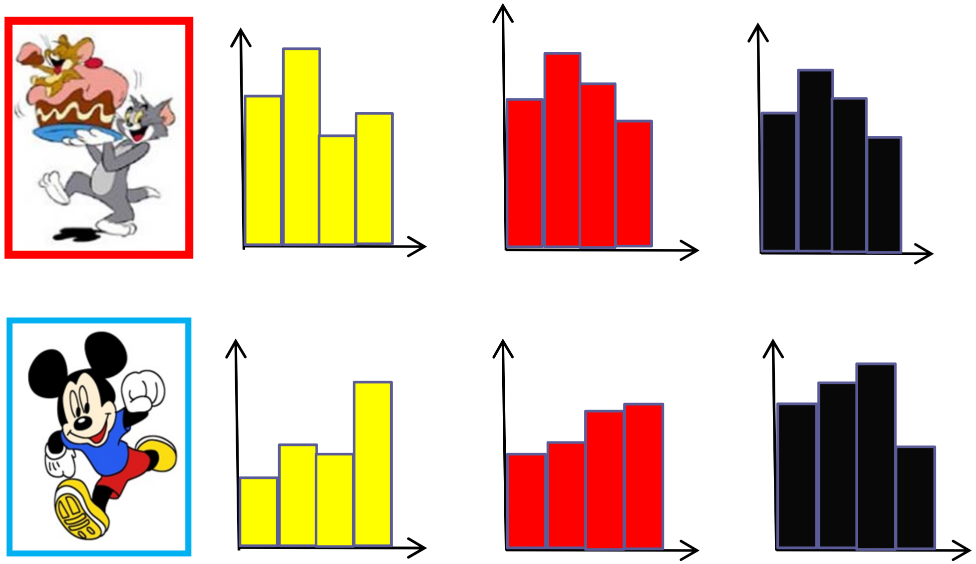
- Mathematical computing through visual descriptors



How to compute image similarity

Traditional Solutions:

- Mathematical computing distance of visual descriptors



Euclidean distance

Earth Mover distance

Jaccard distance

Hamming distance

Mahalanobis distance

Correlation distance

Manhattan distance

Minkowski distance

Hausdorff distance

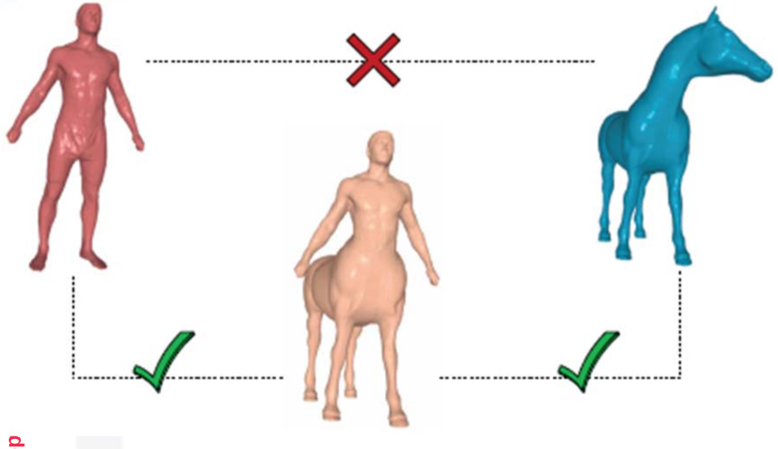
Chebyshev distance

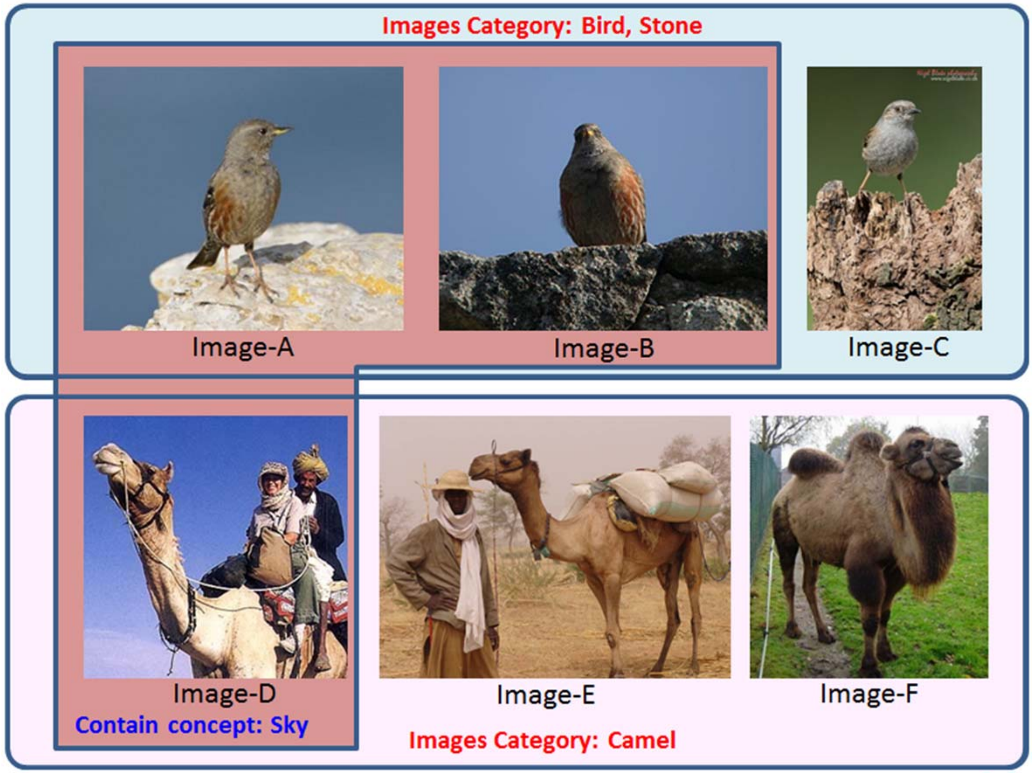
Cosine distance

……

How to compute visual similarity

Non-metric similarity modeling

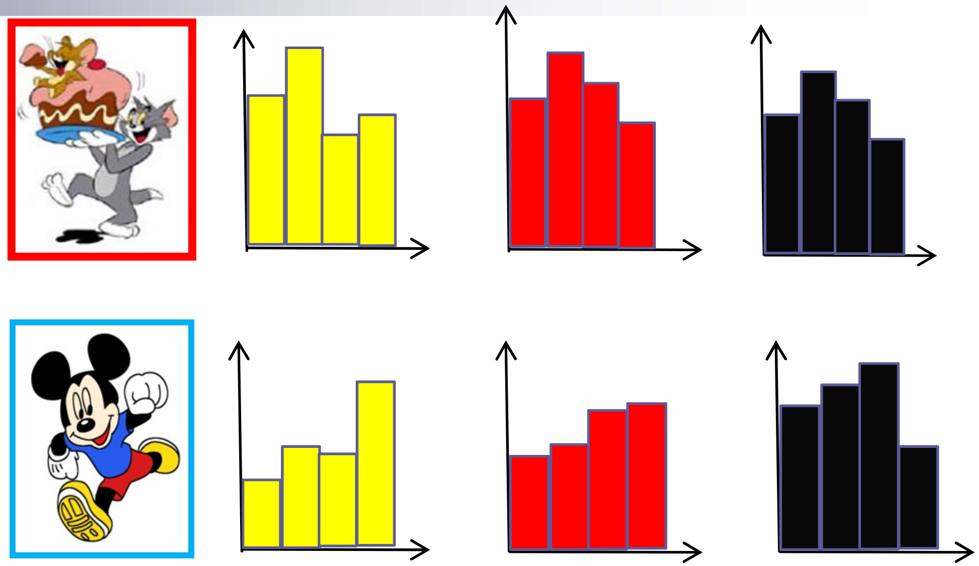




How to compute visual similarity

Traditional Solutions:

- Mathematical computing through visual descriptors

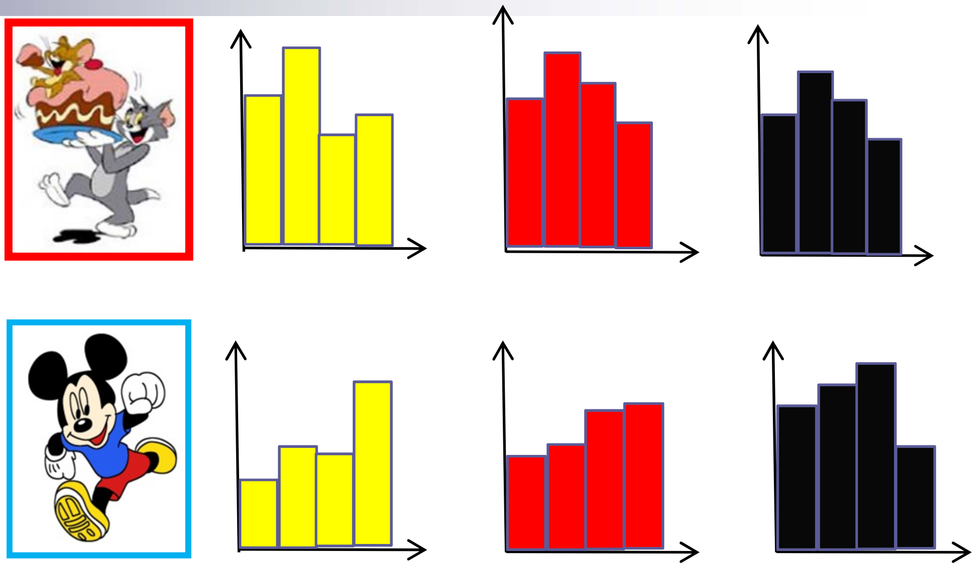


- Disadvantage
  - □ Visual descriptor could not fully represent the original image
  - □ Big gap between human’s recognition and digital computation
  - □ Visual similarity is not consensus among users

How to compute visual similarity

Most Solutions:

- Mathematical computation through visual descriptors

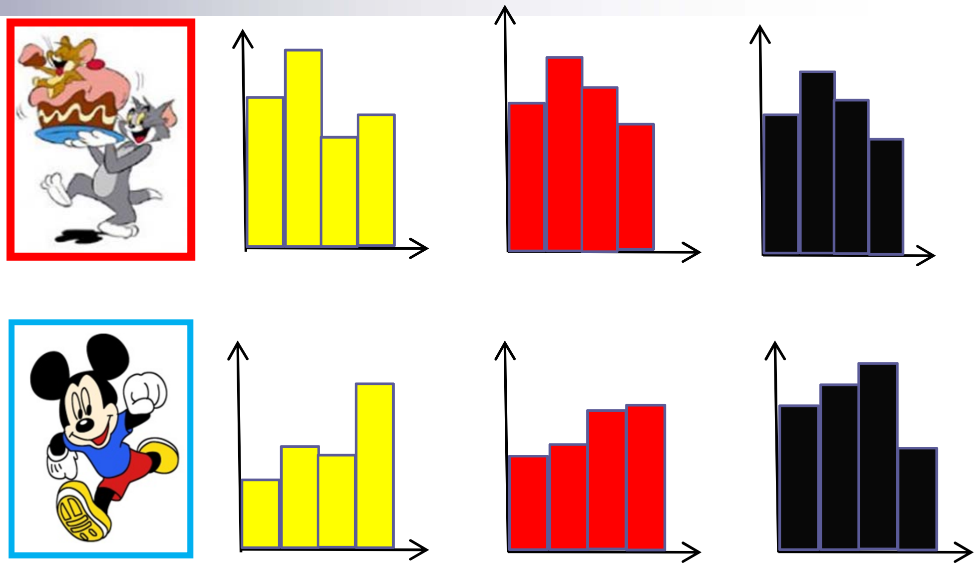


Social information could help!

How to compute visual similarity

Most Solutions:

- Mathematical computation through visual descriptors

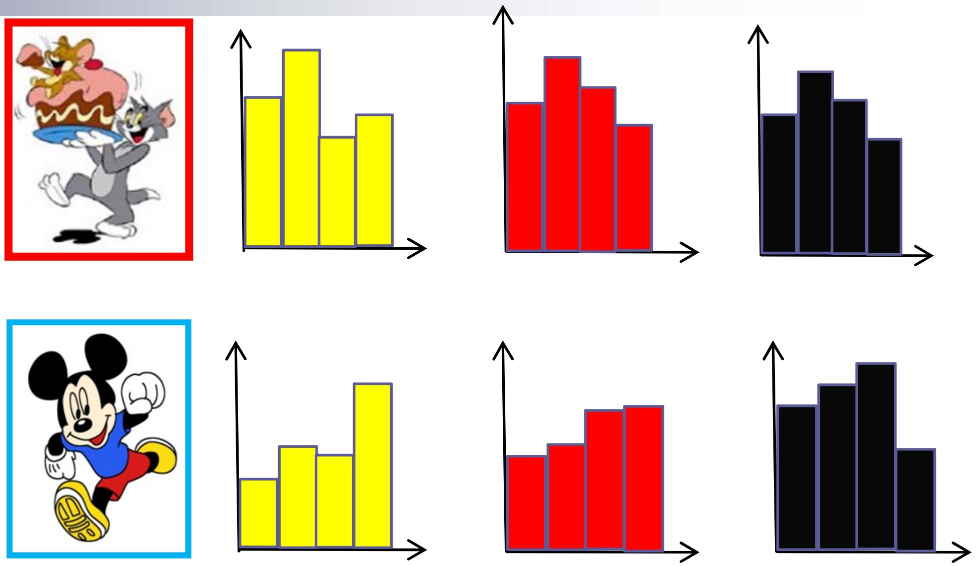


- Disadvantage
  - □ Visual descriptor could not fully represent the original image
    - Textual information in social context is more reliable
  - □ Big gap between human's recognition and digital computation
    - Social information are generated by many people
  - □ Visual similarity is not consensus among users
    - Social information can represent the public opinion in many cases

How to compute visual similarity

Most Solutions:

- Mathematical computation through visual descriptors



Social information could help!

It is also a complex issue !

Many images on the web

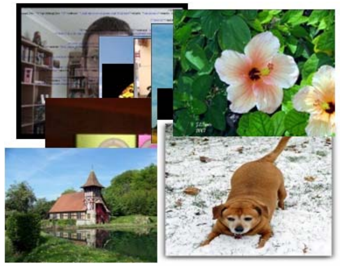
IMAGENET
PASCAL2
Pattern Analysis, Statistical Modelling and Computational Learning

Well labeled images

flickr

sky
sunset
lake
sea
tree

Noisy labeled Images



Unlabeled images

Many images on the web

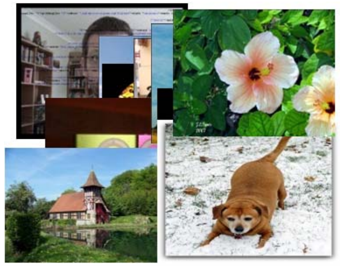
IMAGENET
PASCAL2
Pattern Analysis, Statistical Modelling and Computational Learning

Well labeled images

flickr

sky
sunset
lake
sea
tree

Noisy labeled Images



Unlabeled images

Social Activity

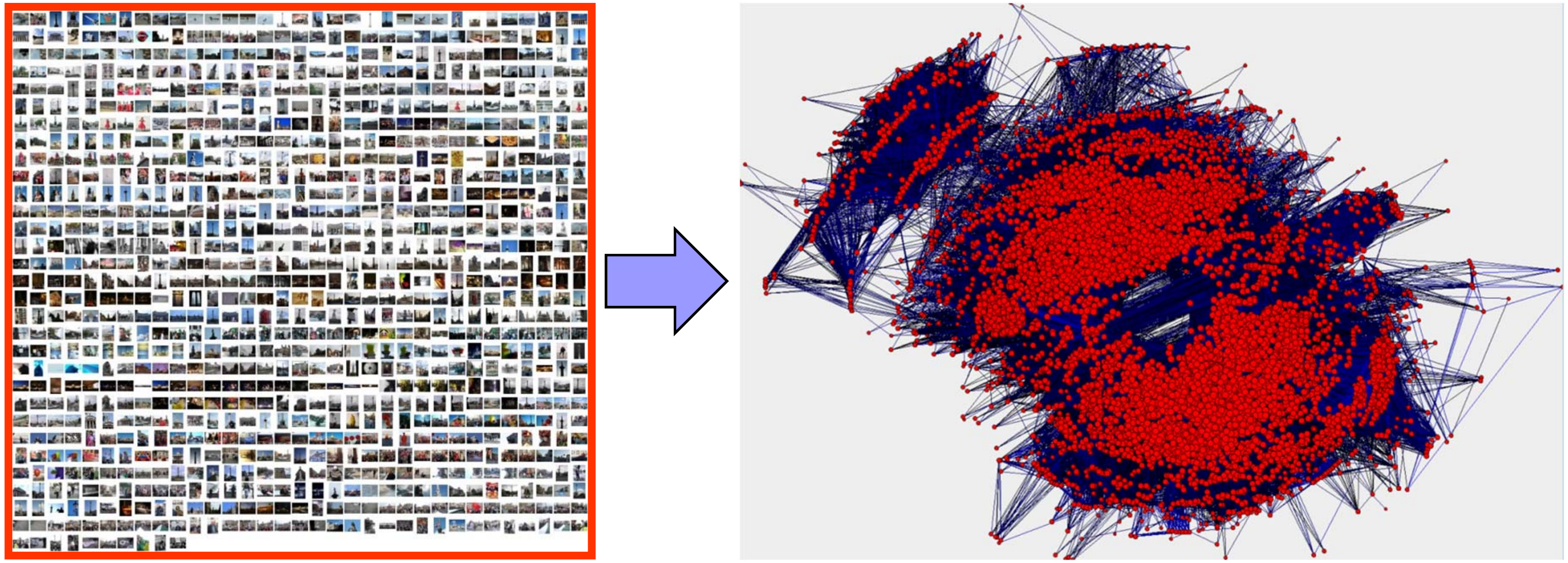
Social Connection

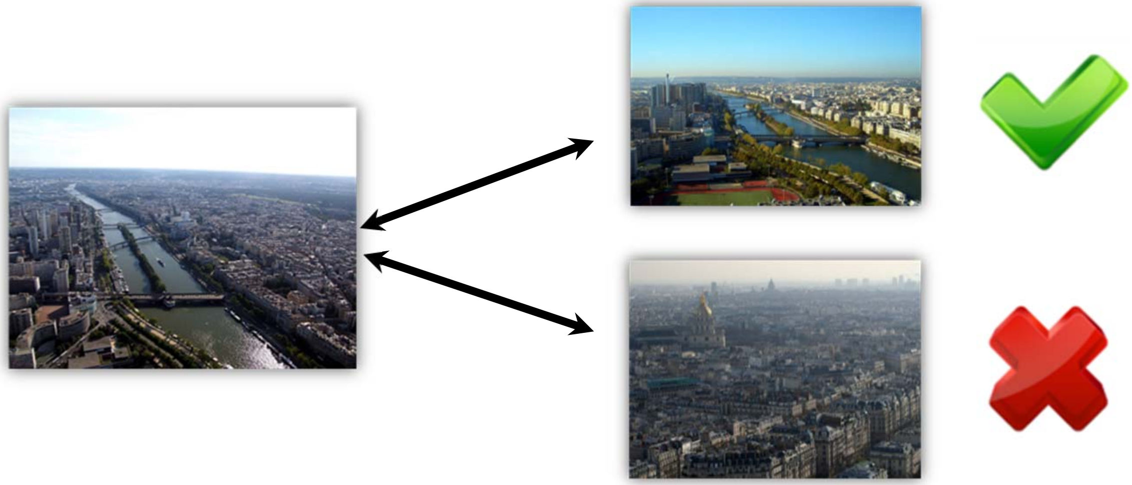
Social Platform

中科院计算所

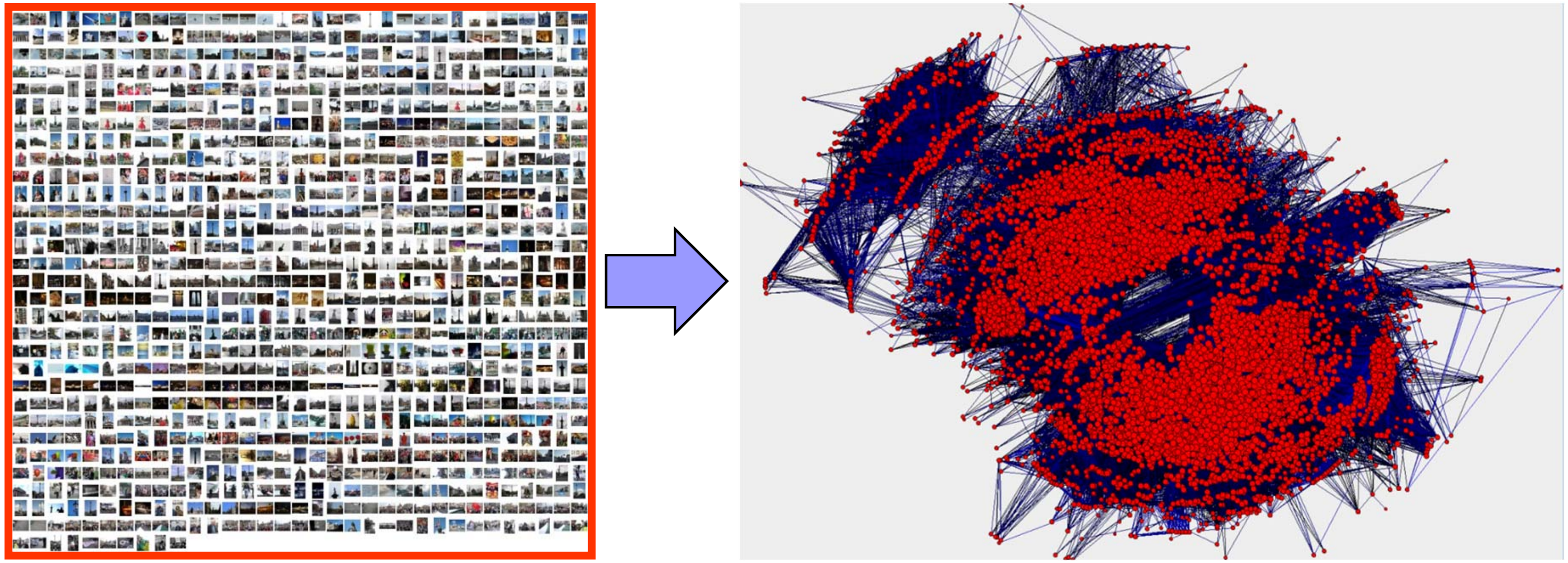
Institute of Computing Technology, Chinese Academy of Sciences

Computing image similarity





Computing image similarity

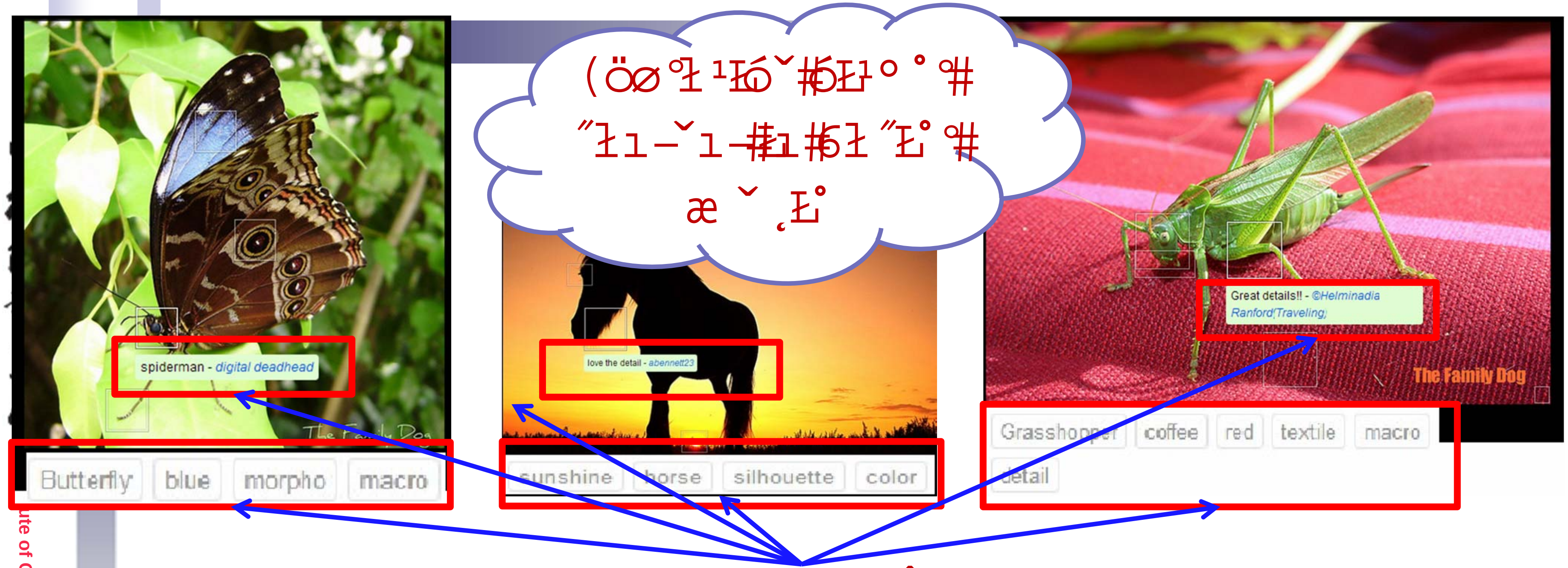


Visual descriptor + Social information

Some techniques

- Image similarity with social tags
- Image similarity with hierarchical semantic relations

Visual Content in Social Media



(Öø ºł ¹Łő ˇ#6Ł¹° ˚ º#
˝łı–ˇı–#ı #6ł ˝Ł˚ º#
æ ˇ ˛Ł˚

7ª ˇ#ł ˝Ł˚ º#–˚ÆÆŁıÆ#ı–łßæ ˚–Łłı#˝łı–ßŁ¸ °–ˇ ˛#¸Ê#º˚ßÆˇ#ı °æ ¸ ˇß#ł–#º ¹ˇß¹

A. The users give the tagging freely, so it contains a lot of noise.

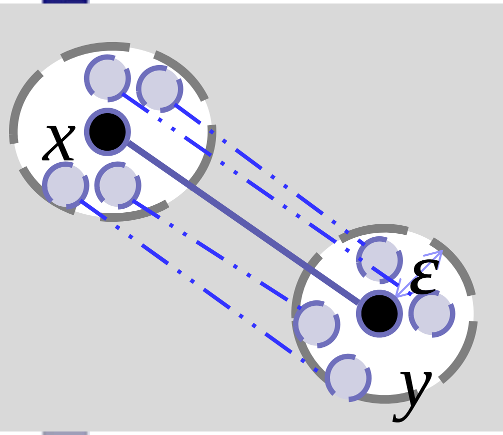
B. It is provided by many users, so it is abundant and contains subjective intention.

How can we take advantage of social tagging for visual content analysis

A. Use them in a noise-resistant manner.

B. Use them as an auxiliary information for model learning.

Neighborhood Similarities



- Basic assumptions：
  - □ Data on regions with similar local density is more similar than data on regions with different local density.
  - □ Data on dense manifolds tend to be more similar than sparse manifolds.

Neighborhood Similarity:

$$K_N(\mathbf{x},\mathbf{y}) = \alpha K_O(\mathbf{x},\mathbf{y}) + (1-\alpha)\frac{\sum K_O(\mathbf{x}',\mathbf{y}')}{|Nbd(\mathbf{x})||Nbd(\mathbf{y})|}$$

$$\mathbf{x}' \in Nbd(\mathbf{x}),\ \mathbf{y}' \in Nbd(\mathbf{y}),\ \mathbf{x}',\mathbf{y}' \in U$$

- Advantage：
  - It appropriately measures the distance of two convex hulls formulated by two sets of neighborhood data, instead of over-sensitive point-to-point distance.
  - Robust to noise.

Metric Learning and Multiple Feature Fusion

- Conduct distance metric learning(DML) on each feature channel

$$K_L(\mathbf{x},\mathbf{y}) = K(\mathbf{Lx},\mathbf{Ly})$$

$$K_N^{(m)}(\mathbf{x},\mathbf{y}) = \alpha K_L^{(m)}(\mathbf{x},\mathbf{y}) + (1-\alpha)\frac{\sum K_L^{(m)}(\mathbf{x}',\mathbf{y}')}{|Nbd^{(m)}(\mathbf{x})||Nbd^{(m)}(\mathbf{y})|}$$

$$\mathbf{x}' \in Nbd^{(m)}(\mathbf{x}), \mathbf{y}' \in Nbd^{(m)}(\mathbf{y}), \mathbf{x}',\mathbf{y}' \in U$$

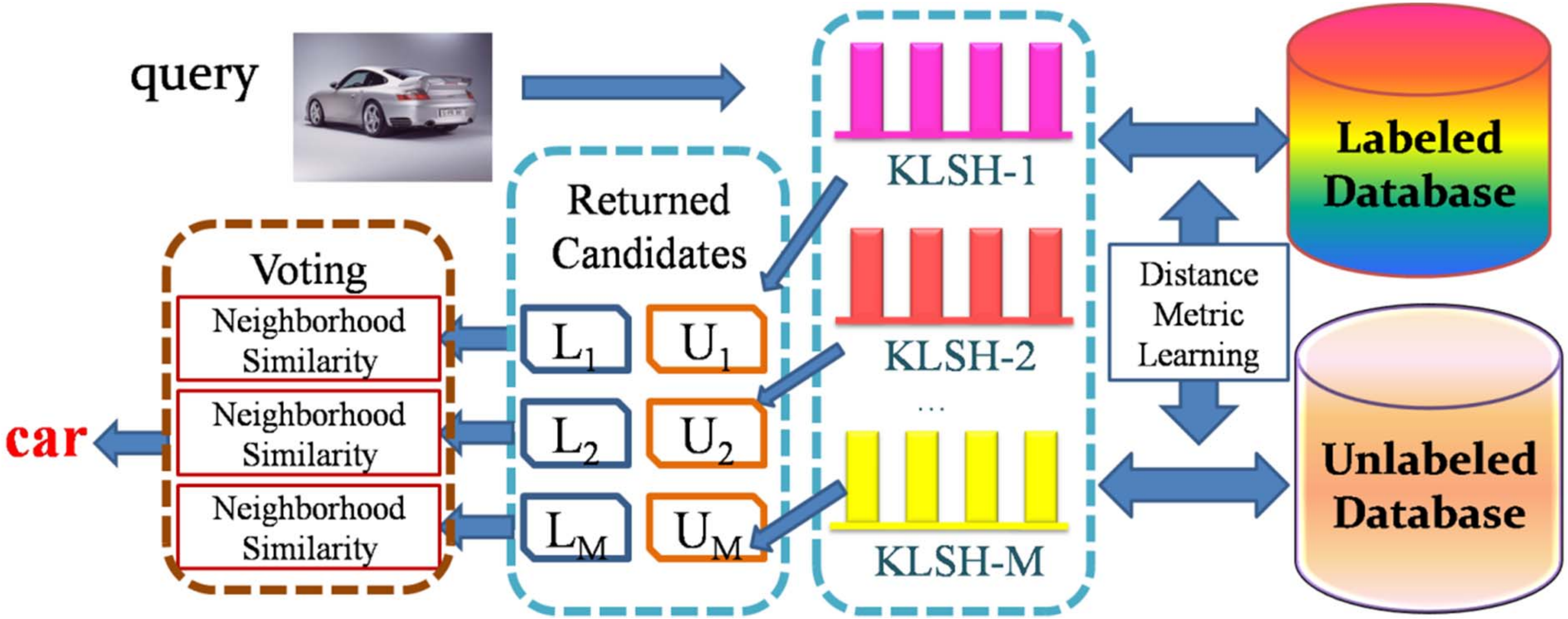
- Fusing multiple features：

$$K_N(\mathbf{x},\mathbf{y}) = \sum_{m=1}^{M} w_m K_N^{(m)}(\mathbf{x},\mathbf{y}),\ \ s.t.\ \ w_m \geq 0, \sum_{m=1}^{M} w_m = 1$$

w_m can be tuned on a given validation set

Framework

- Implementation details towards large scale data：
  - □ Several KLSHs are built on each feature channel.
  - □ We construct 3 hash tables for each KLSH, so that higher recall can be achieved.



Experimental Results(I)

- Dataset

Caltech256:30K

Web images:2M

#features：5

Methods	Performance	Methods	Performance
NN-1	33.0 ±2.1%	D-NN-1	37.5±1.8%
NN-3	36.5±1.75%	D-NN-3	41.5±1.6%
NN-5	40.1±1.4%	D-NN-5	**43.6±1.31%**
UNN-1	35.0±1.1%	D-UNN-1	40.1±1.0%
UNN-3	38.6±0.76%	D-UNN-3	**44.9±0.9%**
UNN-5	**44.4±0.42%**	D-UNN-5	**47.1±0.37%**
[Boiman08]	≈42%		

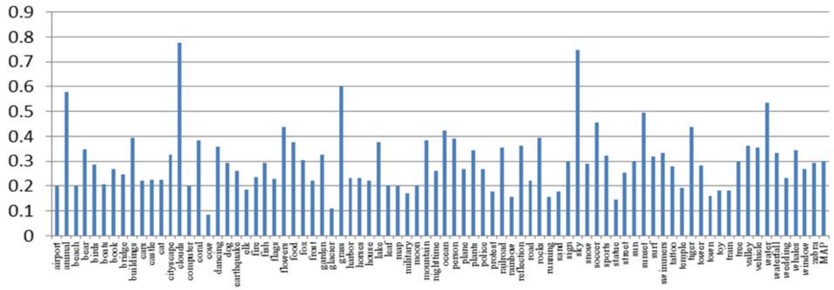
Large scale Web image can help the model to better reflect the true distribution in high dimensional feature space, which can be used in our neighborhood similarity and make it better approximate the true local density information

Average Retrieval Time (Platform: Matlab, in seconds)

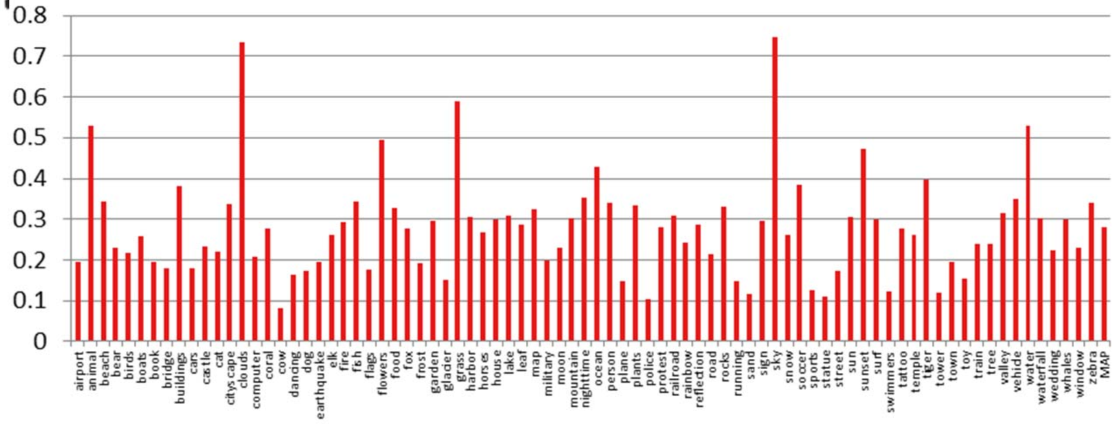
#Neighbors	1	3	5	10	15	20
UNN-5	1.2	1.8	2.6	3.7	5.3	8.8
D-UNN-5	1.3	2.1	2.8	3.9	5.7	9.2

Experimental Results (II)

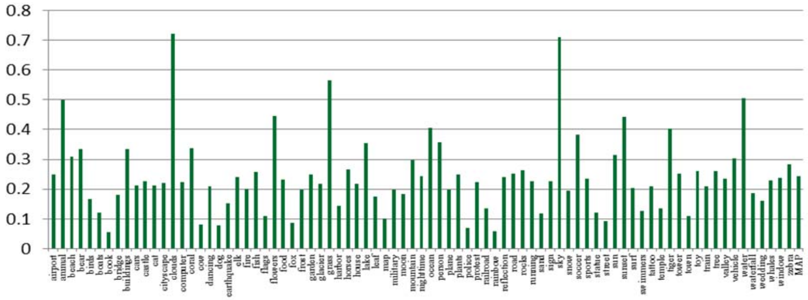
NUS-WIDE Dataset



Using all the labeled training data, MAP: 0.2995



Our approach with 50% labeled data+50% unlabeled data, MAP: 0.2797



Only using 50% labeled data, MAP: 0.2434

Multi-feature metric learning

Motivation： **can we incorporate multiple sources (*i.e.* category information and social tagging) to enhance the semantic consistence of the learned metrics？**

Solution outline： design a multi-task learning framework to learning multiple (hyper-)category specific metrics with information sharing.

The propose metric definition：

$$\tilde{K}_t^{ij} = \sum_{m=1}^{M} \tilde{K}_t^{ij,m}, \quad \tilde{K}_t^{ij,m} = (x_t^{i,m})^* \boxed{\Big(A_0^{(m)}} + A_t^{(m)}\Big) x_t^{j,m}$$

$$\tilde{d}_t^{ij} = \sum_{m=1}^{M} \tilde{d}_t^{ij,m}, \quad \tilde{d}_t^{ij,m} = (x_t^{i,m} - x_t^{j,m})^* \boxed{\Big(A_0^{(m)}} + A_t^{(m)}\Big)(x_t^{i,m} - x_t^{j,m})$$

A_0 denotes the shared metric in our multi-task metric learning framework

The primal problem based on ideal kernel, *l*$_p$-MKL and MTL:

$$\min_{\mathbf{b},\mathbf{A}} \boxed{\frac{1}{2}\left(\gamma_0 \sum_{m=1}^{M} \frac{1}{b_0^{(m)}} \| A_0^{(m)} \|_F^2 + \sum_{t=1}^{T}\sum_{m=1}^{M} \frac{\gamma_t}{b_t^{(m)}} \| A_t^{(m)} \|_F^2 \right)} + \boxed{\frac{C}{N}\sum_{t=1}^{T}\sum_{ij\in S} \xi_t^{ij}} + \boxed{\frac{\eta}{2}\sum_{t=0}^{T} \| \mathbf{b}_t \|_p^2}$$

$$s.t. \quad \delta_t^{ij}\left(d_t^{ij} - \tilde{d}_t^{ij}\right) \geq \sigma_t^{ij} - \xi_t^{ij}, \ \ \xi_t^{ij} \geq 0, b_t^{(m)} \geq 0, p > 1, \ \ A_t^{(m)} \succeq 0$$

Regularization on *A*

Empirical loss

Regularization on Kernel weight

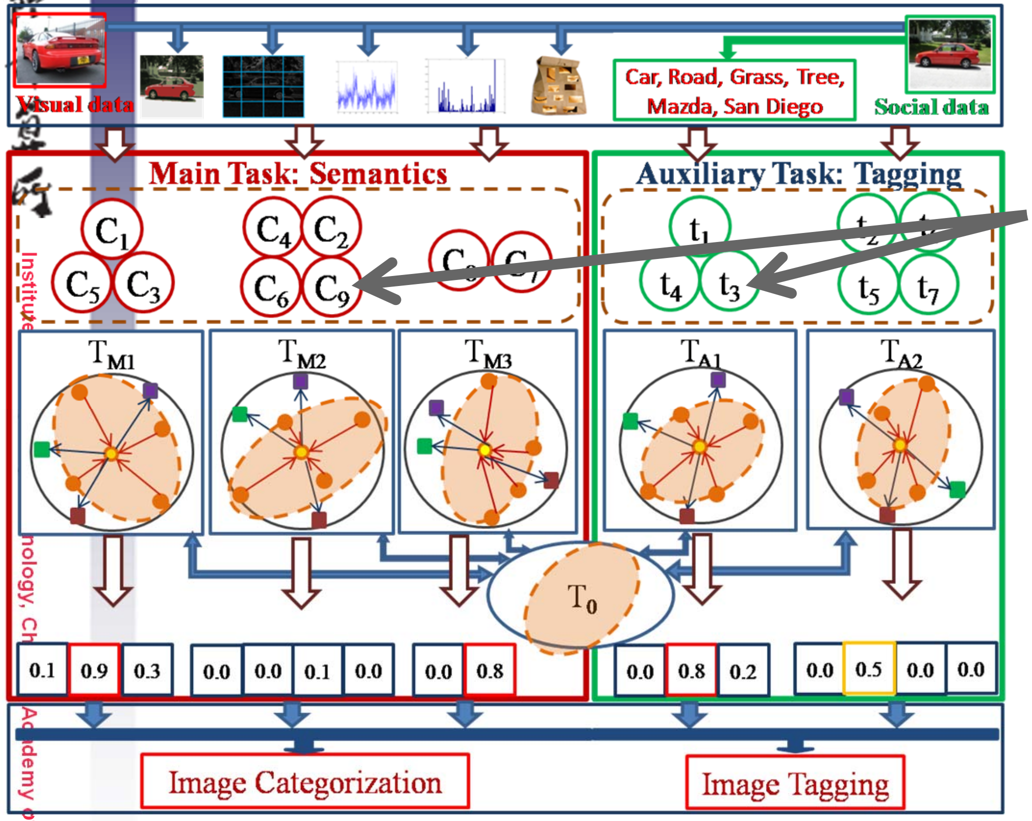
The dual problem is smooth convex function：

$$D: \min_{\boldsymbol{\alpha}} R(\boldsymbol{\alpha}) = -\sum_{t=1}^{T} \mathbf{s}_t' \boldsymbol{\alpha}_t + \frac{1}{8\gamma_0^2 \eta}\left(\sum_{m=1}^{M}\left(\boldsymbol{\alpha}'\mathbf{Q}^{(m)}\boldsymbol{\alpha}\right)^q\right)^{\frac{2}{q}} + \sum_{t=1}^{T}\frac{1}{8\gamma_t^2\eta}\left(\sum_{m=1}^{M}\left(\boldsymbol{\alpha}_t'\mathbf{Q}_{t,t}^{(m)}\boldsymbol{\alpha}_t\right)^q\right)^{\frac{2}{q}}$$

$$s.t. \ \forall \hat{x}_t^{ij} \in S: 0 \leq \alpha_{ij}^t \leq \frac{C_S}{N_S} \quad \forall \hat{x}_t^{ij} \in D, 0 \leq \alpha_{ij}^t \leq \frac{C_D}{N_D}$$

Learning Framework

Advantage: multiple tasks share information in a unified shared task. The task of semantic categorization(main task) can borrow abundant social tagging information, and the learning task of automatic tagging (auxiliary task) can borrow clean semantic category information .

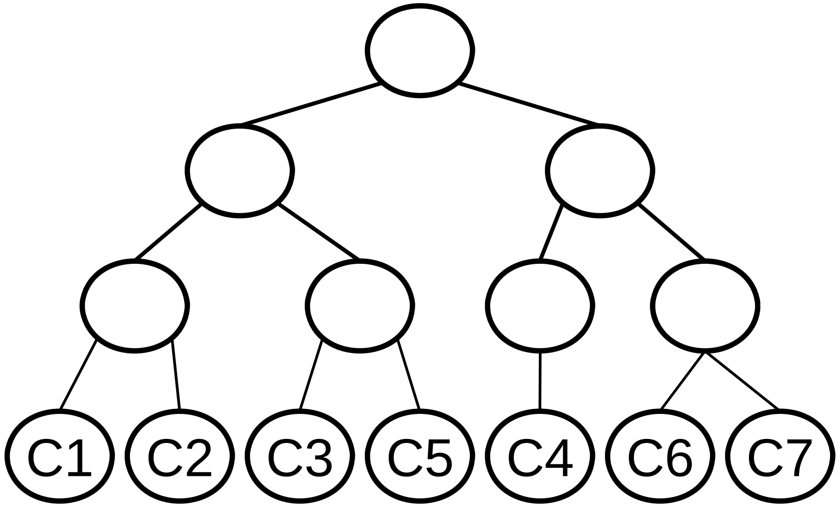


Data： VOC’07：10K

ImageNet-250：250K(250 classes)

MIRFLICKR: 1M

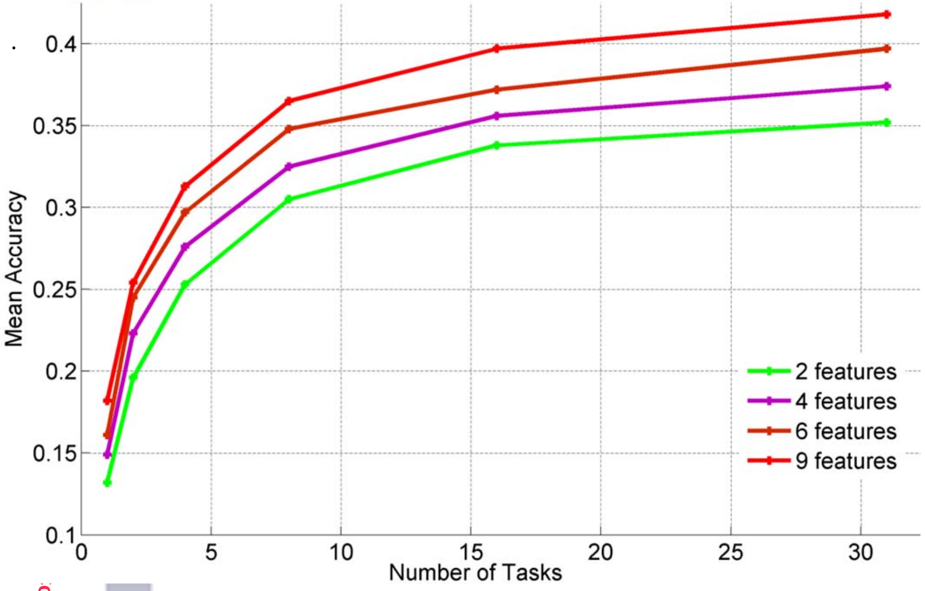
Task grouping based on visual clustering



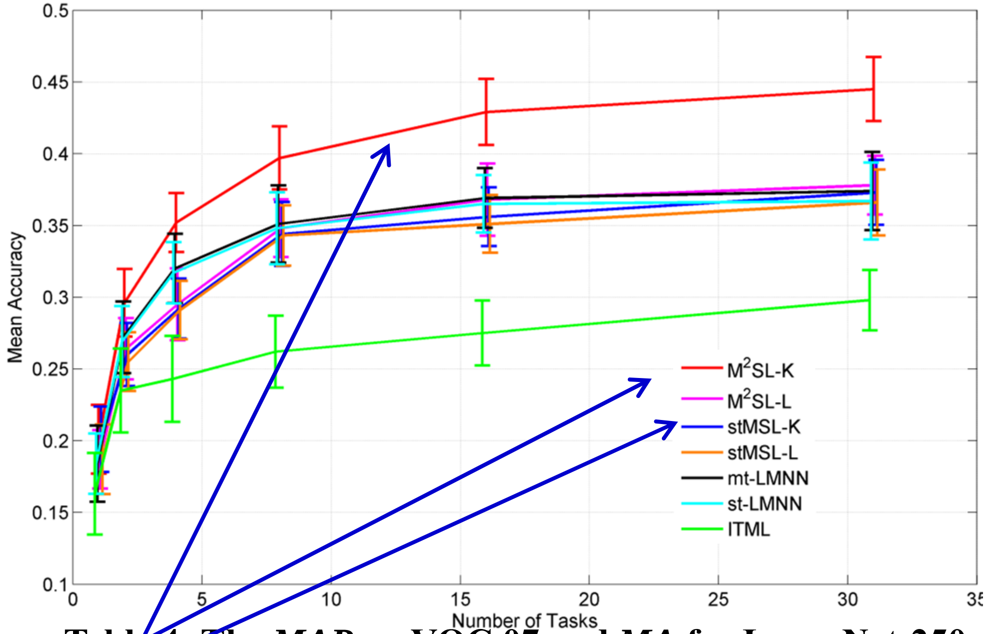
Disadvantage: the proposed task grouping method does not full develop the relation between of hierarchical category level similarity and multi-task learning

Performance of visual categorization

MAP with different #main tasks(M^{2}SL-K)



Comparison with state-of-the-art



Setting: *p*=2.5, $\gamma_t = 1$ $\gamma_0 = 2$ $\frac{C_S}{N_S} = 8$ $\frac{C_D}{N_D} = 4$

Model：Metric learning *k*-NN

A. When the number of categories is large, multi-task learning outperforms single task learning

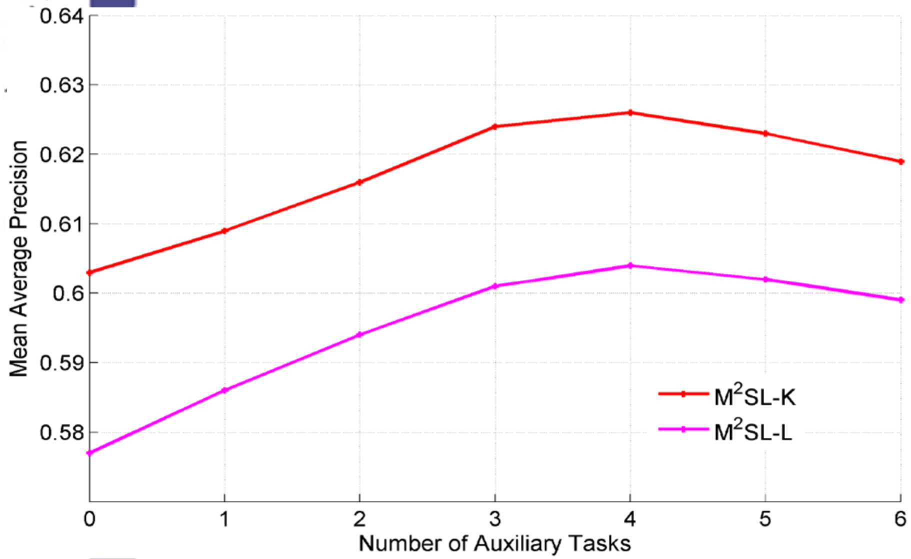
B. Nonlinear metric learning outperforms single task learning

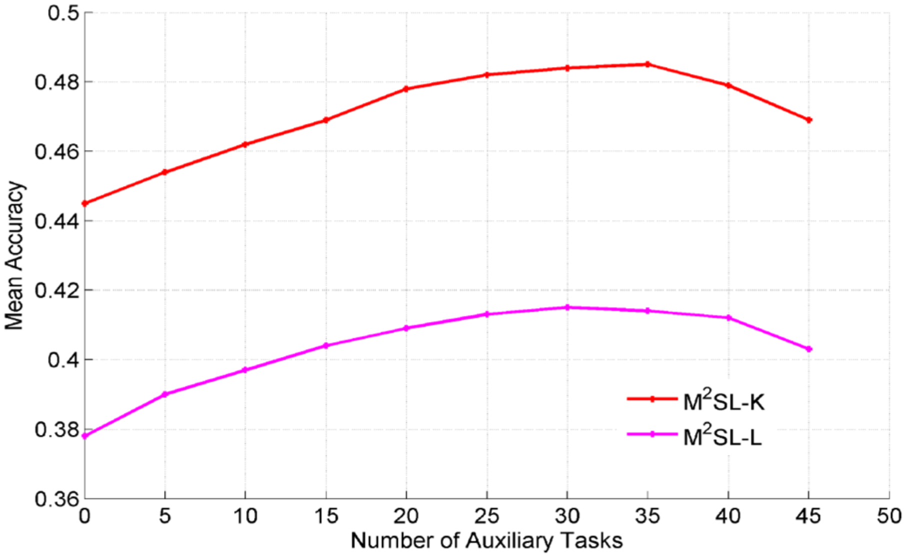
Table 4: The *MAP* on VOC 07 and *MA* for ImageNet-250

Methods	VOC 07	ImageNet-250
EUC	0.181	0.192
EUC-PCA	0.296	0.264
ITML	0.398	0.298
LFDA	0.364	0.305
st-LMNN	0.569	0.367
mt-LMNN	0.572	0.374
NCA	0.375	0.315
M^{2}SL-L	0.577	0.378
M^{2}SL-K	**0.603**	**0.445**

How social tagging helps semantic categorization

Given #main_tasks fixed, the performance on semantic categorization is evaluated on different settings of #auxiliary_tasks





Left： VOC 07　　　　**Right： ImageNet-250**

Experimental finding：

Social tagging is beneficial for semantic categorization, but more data with social tagging means more noisy information.

Cooperative Image Annotation && Future work

tower
Eiffel tower
France
Paris
night

sea
beach
gulf
Sea gull
sky

butterfly
butterfly
flower
nature
tree

airplane
airport
Sky
boeing

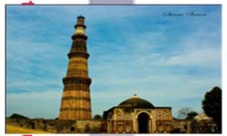
face
Indoor
People
Person

beach
beach
Sky
ocean

ball
kaleidoscope
colors

ipod
apple
mp3
computer

car
race
sports car
Street
person


tower
sky
building
window

train
train
train station
railroad

house
grass
outdoor
tree
bird

wild flower
Flower
garden
plant
nature

wild dog
dog
Snow
wood

sky
sunset
lake
sea
tree

The words in red denotes the results of semantic categorization.

The words in black denotes the results of automatic tagging.

The results shows that our approach provide complementary understanding on visual content.

1st: the model tells more in tagging that it's Eiffel Tower.
14th: the semantic categorization is "wild dog", more accurate than any tag

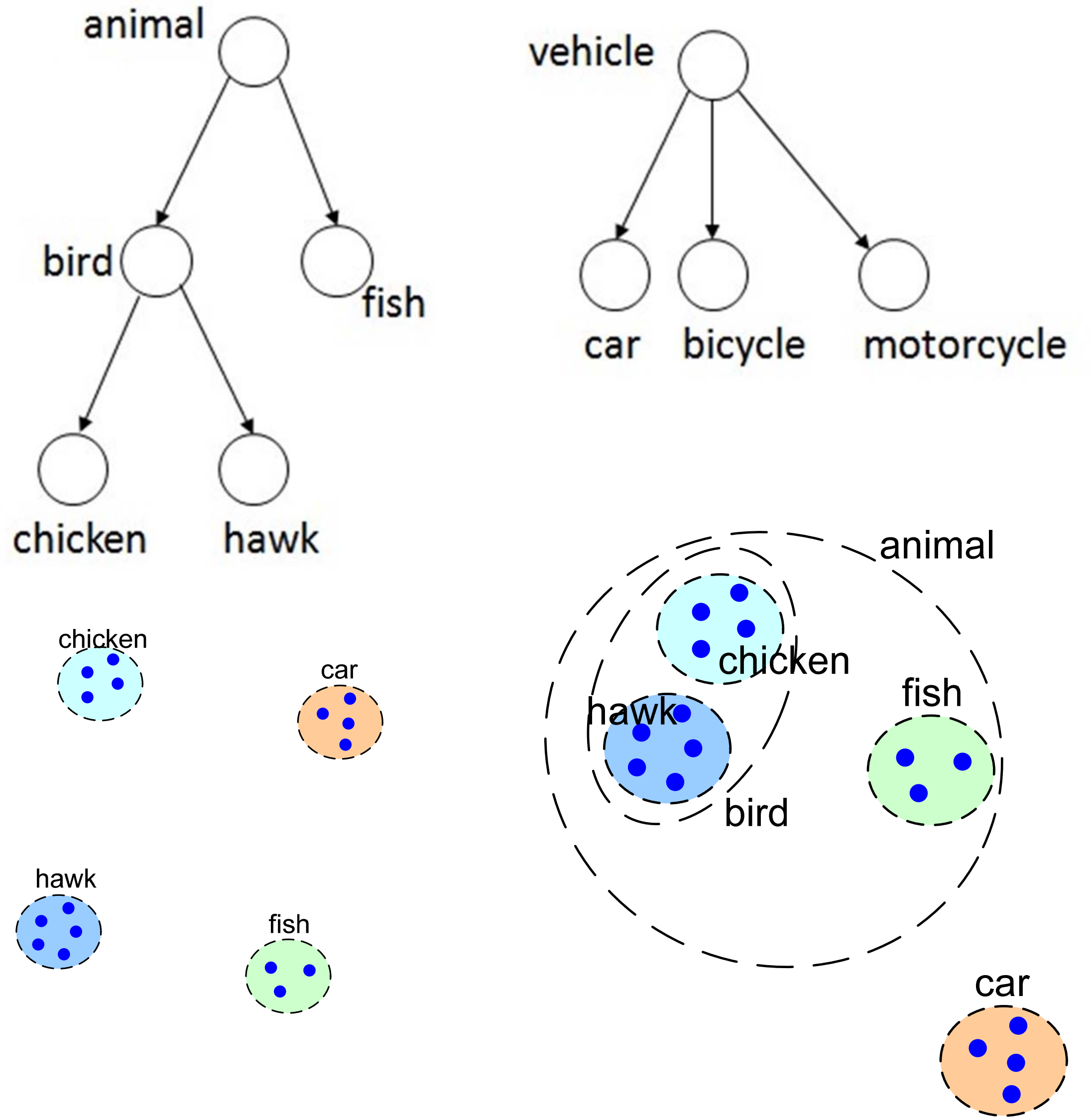
Future work:

We will study how to construct a semantic category structure and use it to provide better information sharing structure for metric learning

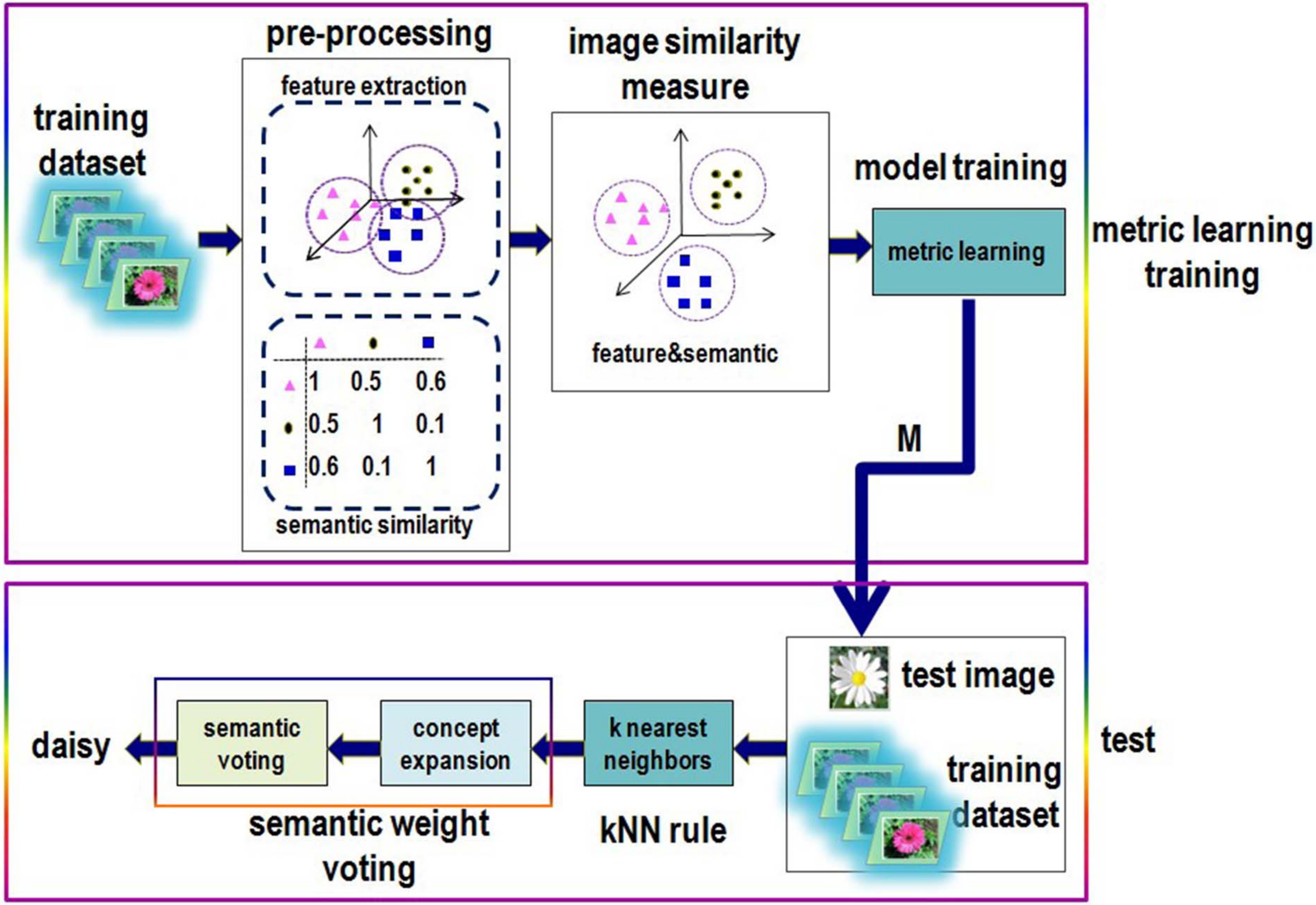
Some techniques

- Image similarity with social tags
- Image similarity with hierarchical semantic relations

Hierarchical Semantic Relations



Proposed Framework

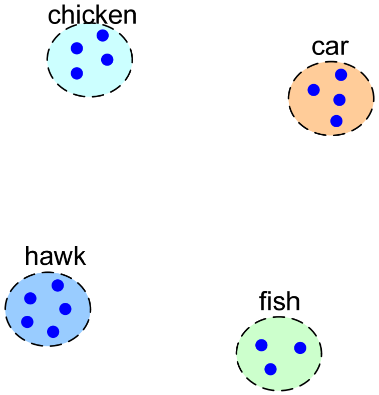


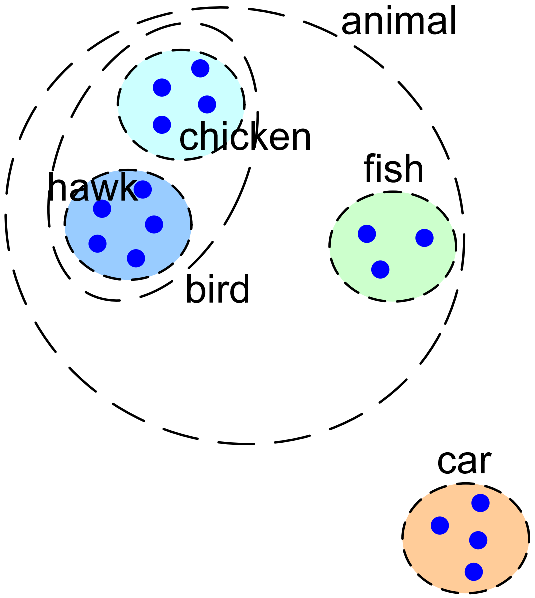
	▲	●	■
▲	1	0.5	0.6
●	0.5	1	0.1
■	0.6	0.1	1

Semantic distance metric learning

中科院计算所

Institute of Computing Technology, Chinese Academy of Sciences

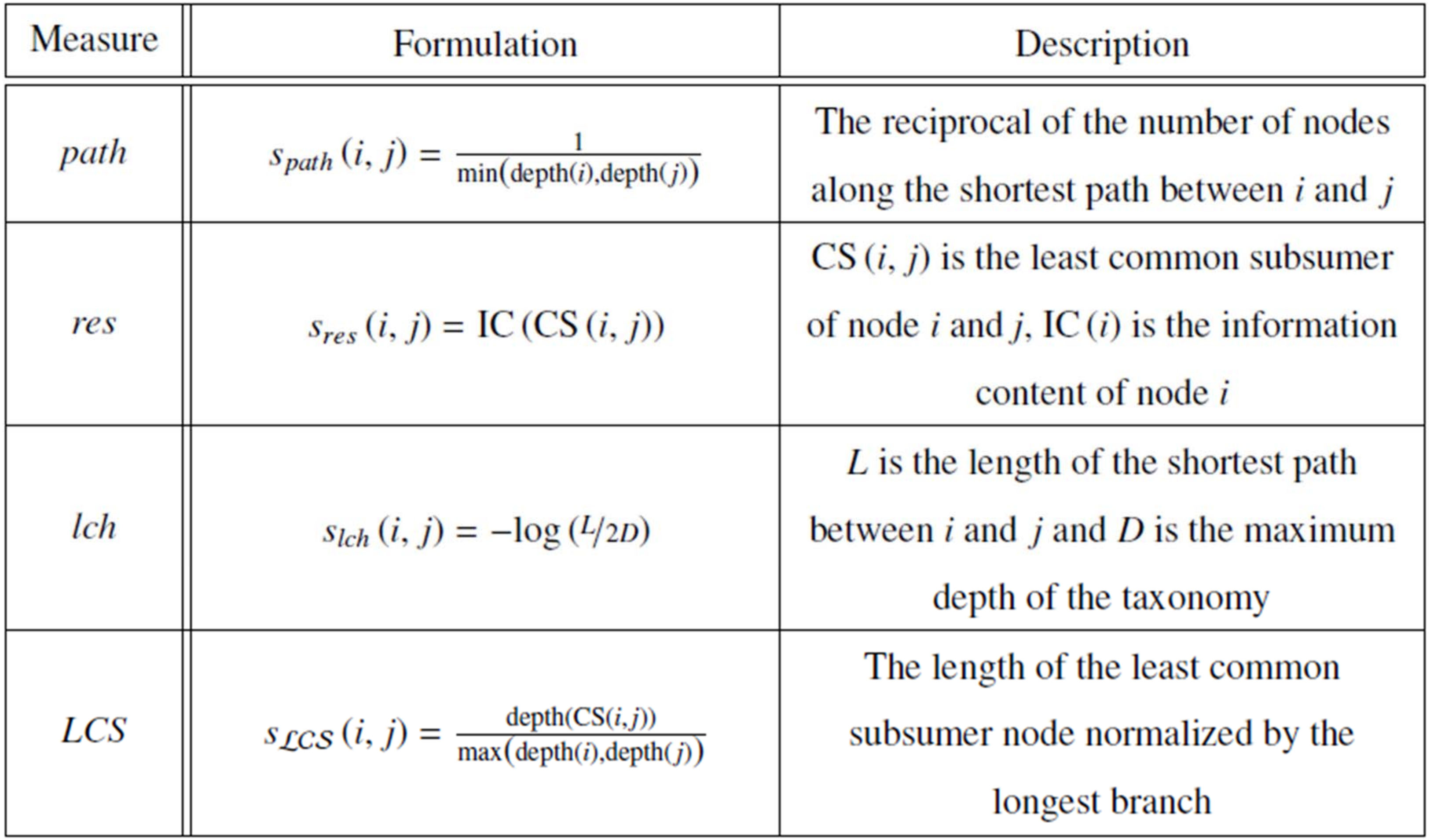




$$I^{(NCA)}(i,j) = v\left(\vec{x}_i,\vec{x}_j\right) = \exp\left(-d^2\left(\vec{x}_i,\vec{x}_j\right)\right) = \exp(-\|A\vec{x}_i - A\vec{x}_j\|^2)$$

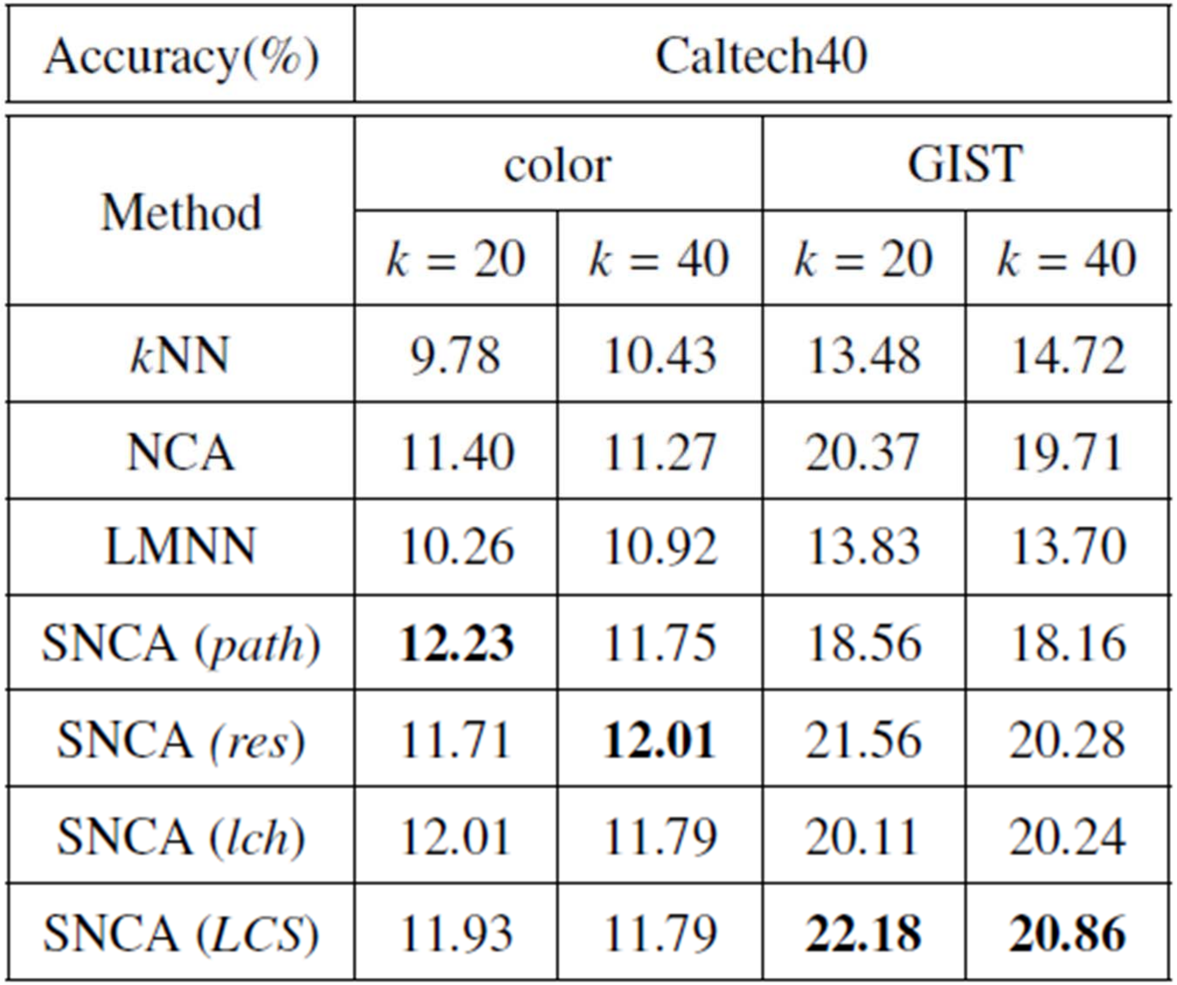
$$I^{(SNCA)}(i,j) = v\left(\vec{x}_i,\vec{x}_j\right) s(i,j) = \exp(-\|A\vec{x}_i - A\vec{x}_j\|^2) s(i,j)$$

Concept similarity measures



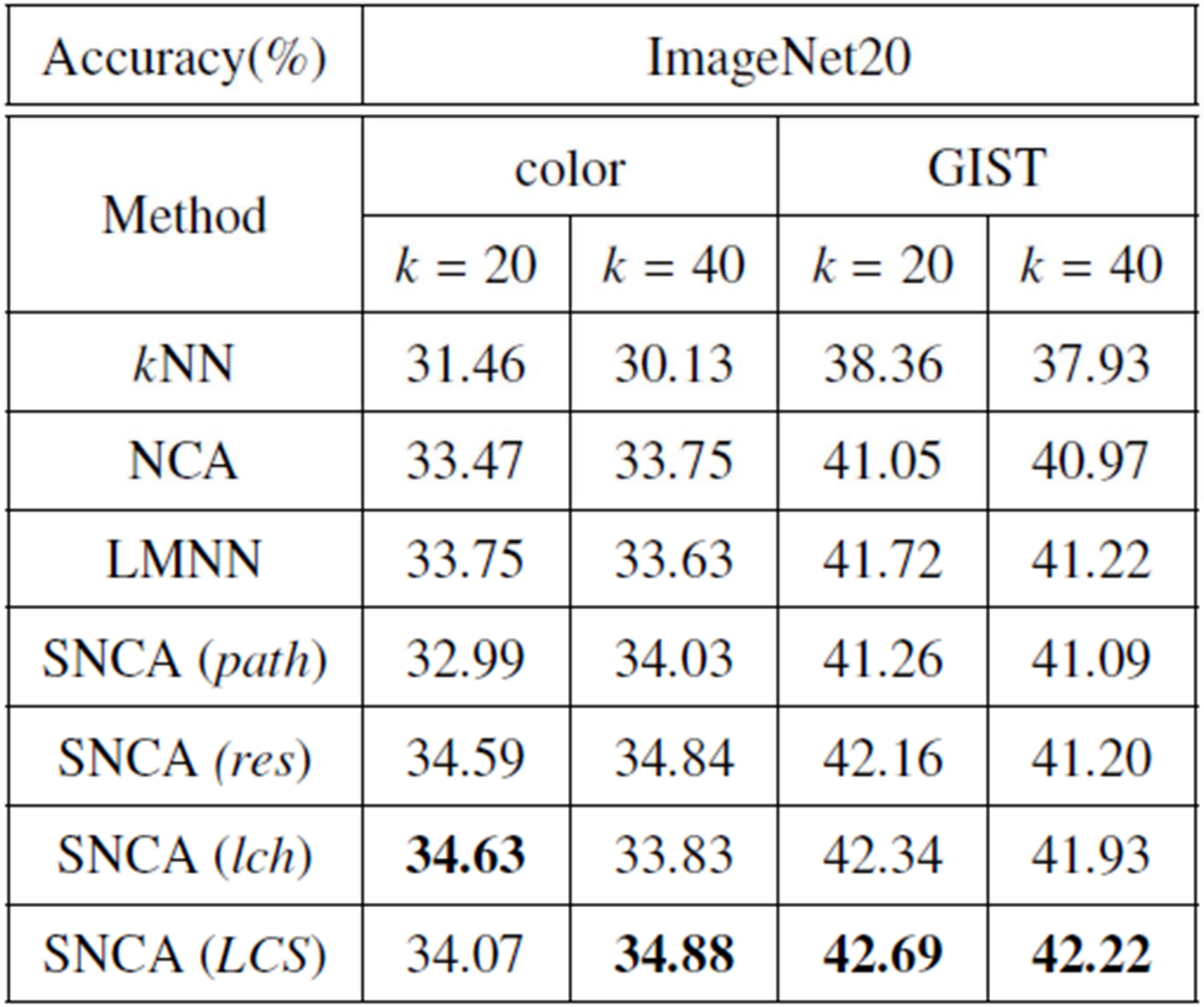
Measure	Formulation	Description
path	$s_{path}(i, j) = \frac{1}{\min(\text{depth}(i),\text{depth}(j))}$	The reciprocal of the number of nodes along the shortest path between *i* and *j*
res	$s_{res}(i, j) = \text{IC}(\text{CS}(i, j))$	CS (*i*, *j*) is the least common subsumer of node *i* and *j*, IC (*i*) is the information content of node *i*
lch	$s_{lch}(i, j) = -\log(L/2D)$	*L* is the length of the shortest path between *i* and *j* and *D* is the maximum depth of the taxonomy
LCS	$s_{LCS}(i, j) = \frac{\text{depth}(\text{CS}(i,j))}{\max(\text{depth}(i),\text{depth}(j))}$	The length of the least common subsumer node normalized by the longest branch

Experimental Results on Caltech40 Dataset



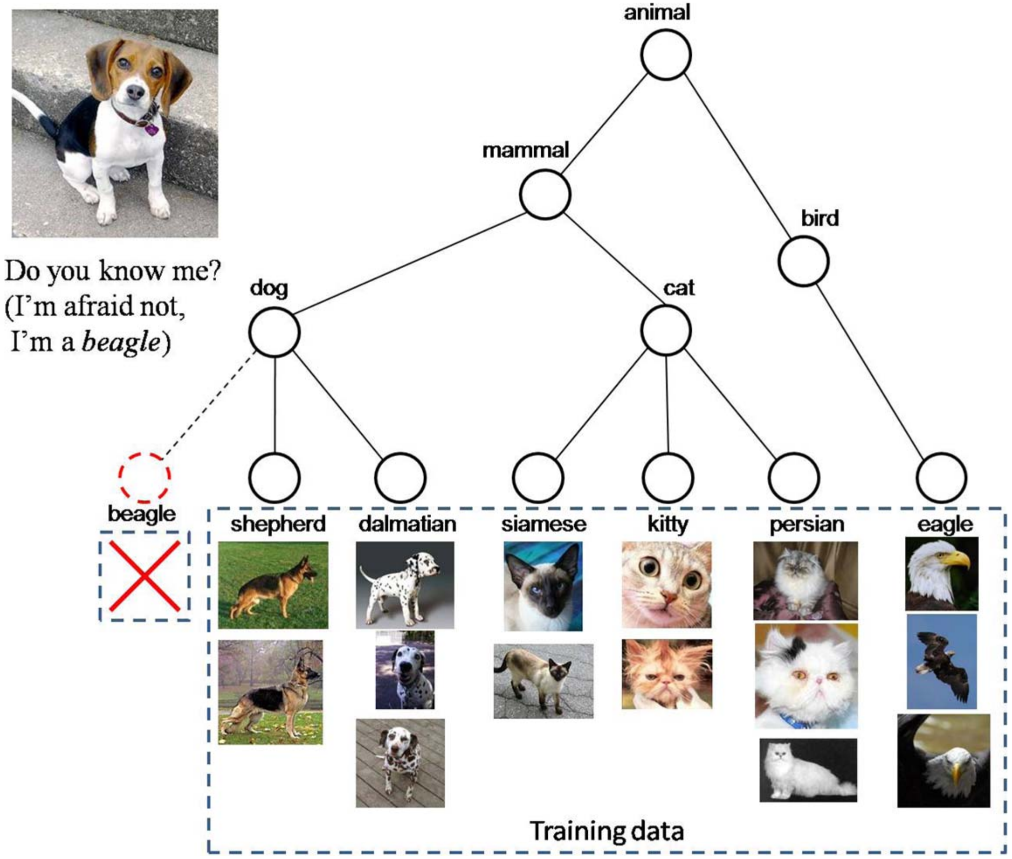
Accuracy(%)	Caltech40			
Method	color		GIST	
	$k = 20$	$k = 40$	$k = 20$	$k = 40$
*k*NN	9.78	10.43	13.48	14.72
NCA	11.40	11.27	20.37	19.71
LMNN	10.26	10.92	13.83	13.70
SNCA (*path*)	**12.23**	11.75	18.56	18.16
SNCA (*res*)	11.71	**12.01**	21.56	20.28
SNCA (*lch*)	12.01	11.79	20.11	20.24
SNCA (*LCS*)	11.93	11.79	**22.18**	**20.86**

Experimental Results on Image40 Dataset

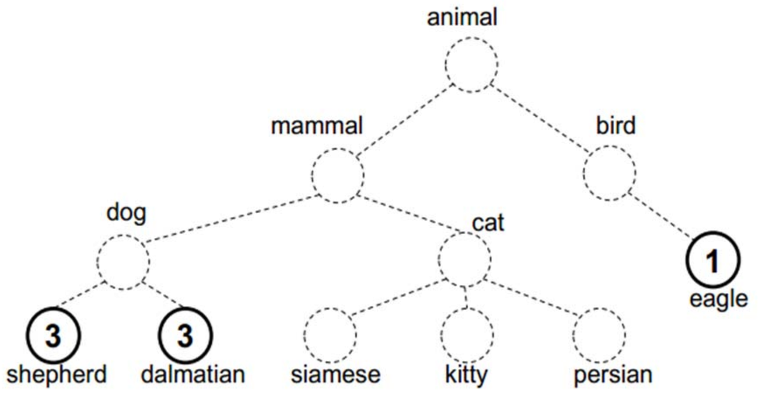


Accuracy(%)	ImageNet20			
Method	color		GIST	
	$k = 20$	$k = 40$	$k = 20$	$k = 40$
*k*NN	31.46	30.13	38.36	37.93
NCA	33.47	33.75	41.05	40.97
LMNN	33.75	33.63	41.72	41.22
SNCA (*path*)	32.99	34.03	41.26	41.09
SNCA (*res*)	34.59	34.84	42.16	41.20
SNCA (*lch*)	**34.63**	33.83	42.34	41.93
SNCA (*LCS*)	34.07	**34.88**	**42.69**	**42.22**

Unknown Concept Annotation

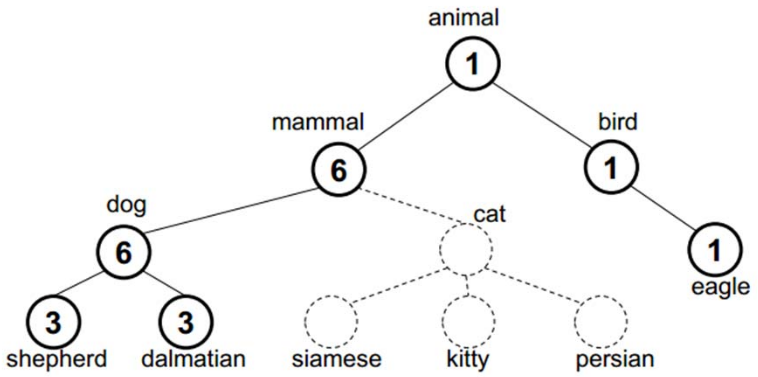


Concept Expansion



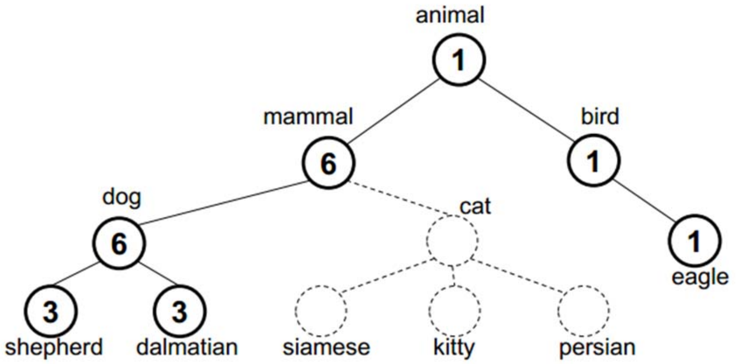
$$CC^{(0)} = \{shepard, dalmatian, eagle\}$$

$$W_{CC}^{(0)} = (3,3,1)$$



$$CC^{(2)} = \{shepard, dalmatian, eagle, dog, bird, mammal, animal\}$$

$$W_{CC}^{(2)} = (3,3,1,6,1,6,1)$$



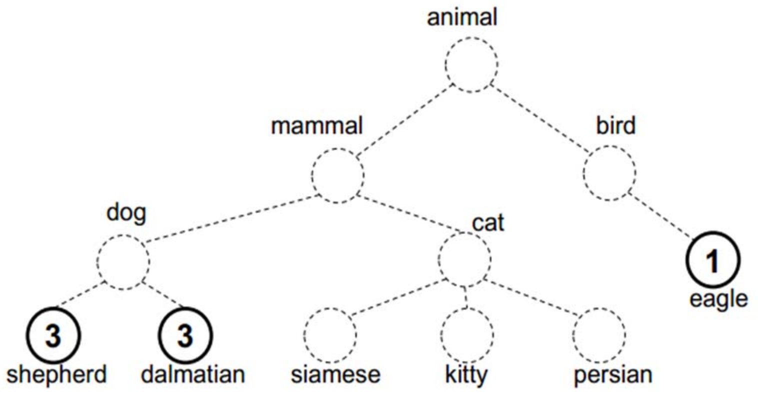
$$CC^{(2)} = \{shepard, dalmatian, eagle, dog, bird, mammal, animal\}$$

$$W_{CC}^{(2)} = (3,3,1,6,1,6,1)$$

中科院计算所

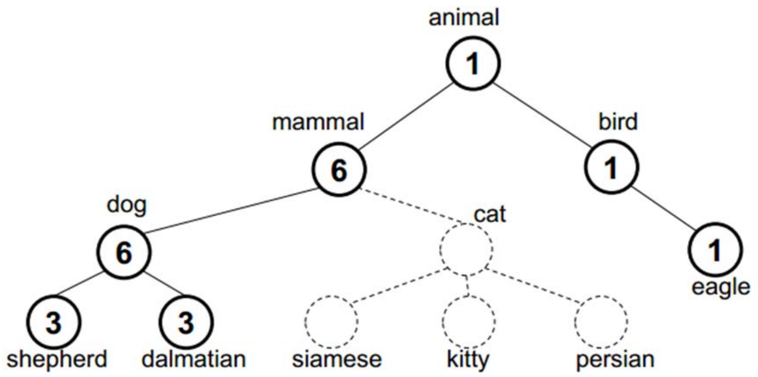
Institute of Computing Technology, Chinese Academy of Sciences

Semantic Voting



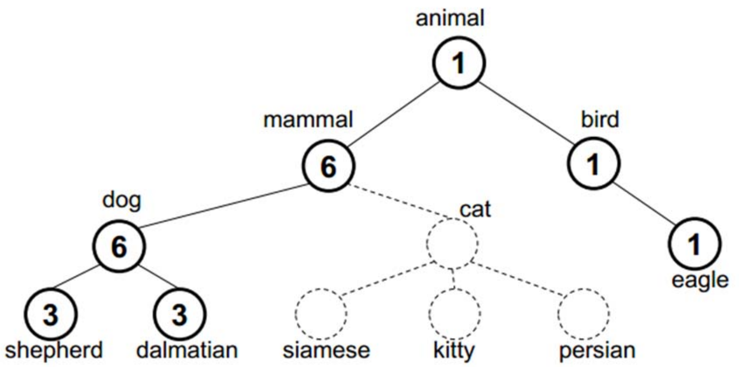
$CC^{(0)} = \{shepard, dalmatian, eagle\}$

$W_{CC}^{(0)} = (3,3,1)$



$CC^{(2)} = \{shepard, dalmatian, eagle, dog, bird, mammal, animal\}$

$W_{CC}^{(2)} = (3,3,1,6,1,6,1)$



$CC^{(2)} = \{shepard, dalmatian, eagle, dog, bird, mammal, animal\}$

$W_{CC}^{(2)} = (3,3,1,6,1,6,1)$

Candidate concept: $CC = \{c_1, c_2, ...c_M\}$

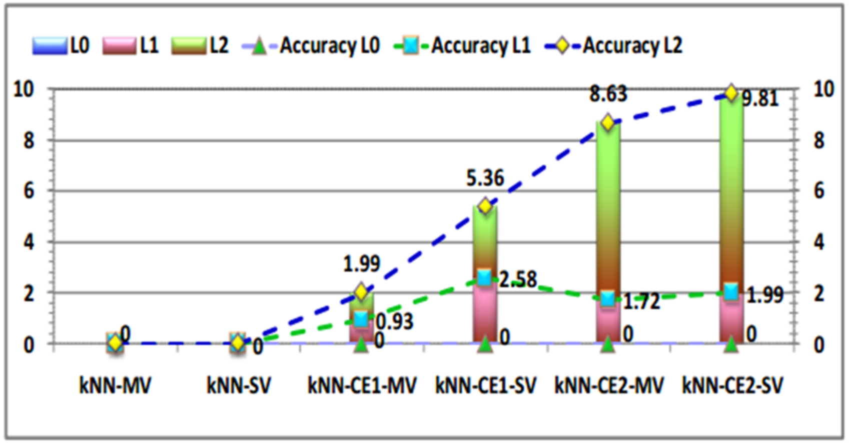
Concept histogram: $W_{CC} = \{w_1, w_2, ...w_M\}$

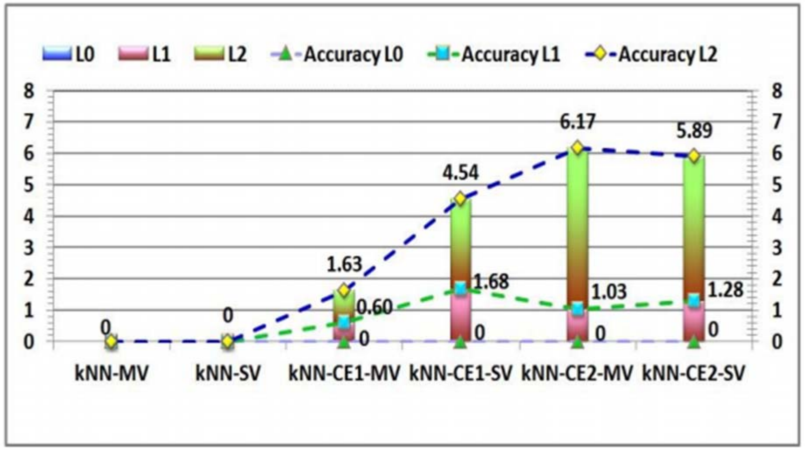
Semantic voting:

$$h(c_i) = \sum_{c_j \in CC} w_j S(c_i, c_j)$$

Experimentation on unknown concept annotation

- GIST and HSV feature with semantic similarity(path)

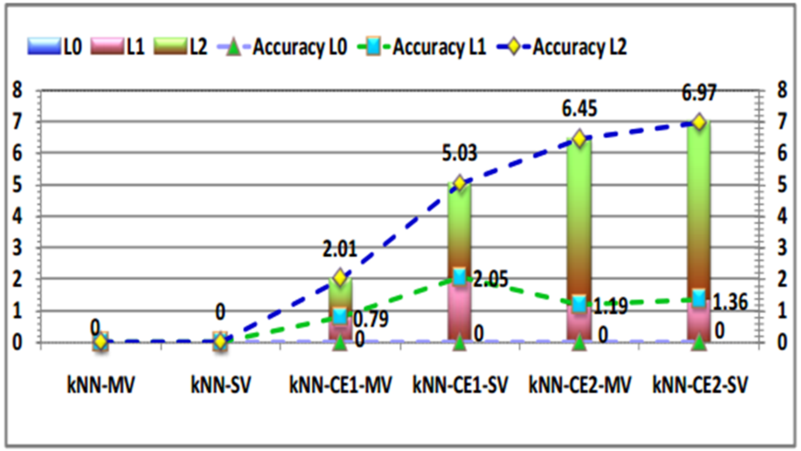


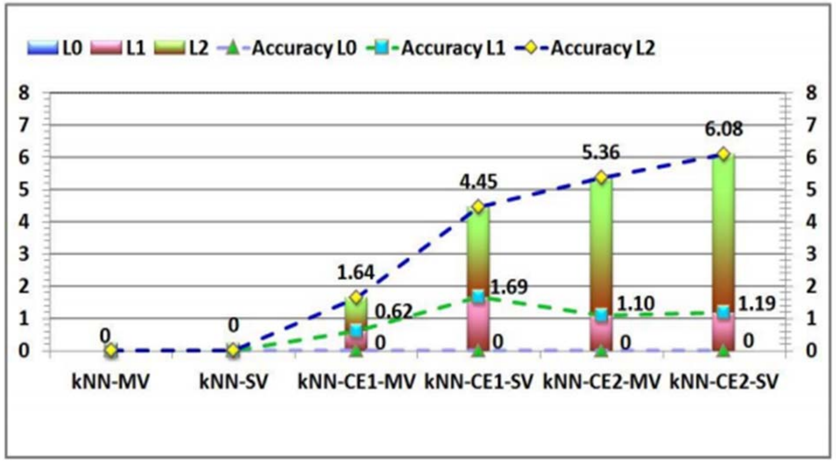


- SV(semantic voting) outperforms MV(majority voting)
- CE(concept expansion) outperforms non-CE

Experimentation on unknown concept annotation

- CM and pHOG feature with semantic similarity(path)





- SV(semantic voting) outperforms MV(majority voting)
- CE(concept expansion) outperforms non-CE

Conclusion

- Image similarity is useful in real applications
- It is a complex and challenging problem
  - □ Only visual information
  - □ Only Social information
  - □ Combining visual and social information together
- Social context information and big data provide a opportunity to satisfactorily solve the problem
  - □ It is still at the preliminary stage, needs a long way to go.

Thanks!