# Tag Suggestion and Localization in User-generated Videos based on Social Knowledge

Lamberto Ballan, Marco Bertini, Alberto Del Bimbo, Marco Meoni, Giuseppe Serra

**ACM SIGMM International Workshop on Social Media (WSM)**
*25 October 2010, Firenze, Italy*

# Worldwide social websites for media sharing

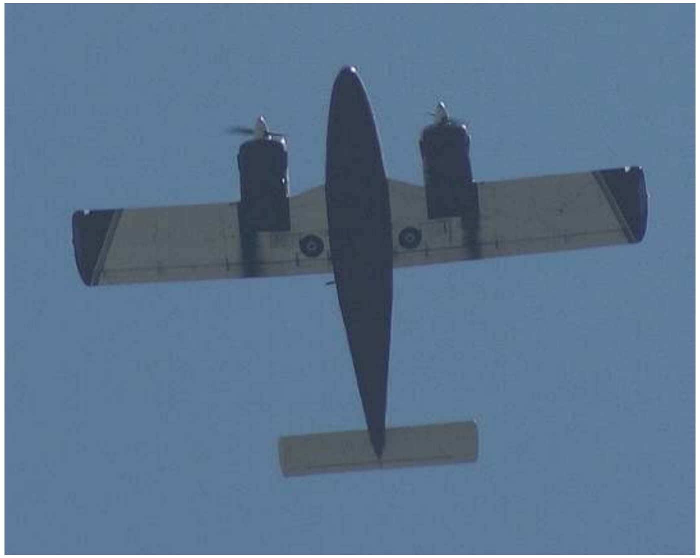
- Social websites for media sharing have become more and more popular in the last years
  - Flickr hosts more than 2 billion images with ~3 millions new uploads per day
  - YouTube reported in March 2010 more than 2 billion views a day and 24 hours of videos uploaded per minute

- People upload, share and annotate multimedia content with ***tags***

# Key problem: social tag reliability

- The performance of social image and video retrieval systems strictly depends on the availability and quality of tags
- But recent studies show that tags are *few*, *imprecise*, *ambiguous* and *overly personalized* [Kennedy *et al.* 2006]
  - e.g. a study on 52 million Flickr photos shows that ~64% of them have only 1-3 tags (see [Sigurbjörnsson and van Zwol 2008] )
- Moreover tags might be irrelevant to the visual content

Query tag: ***airplane***

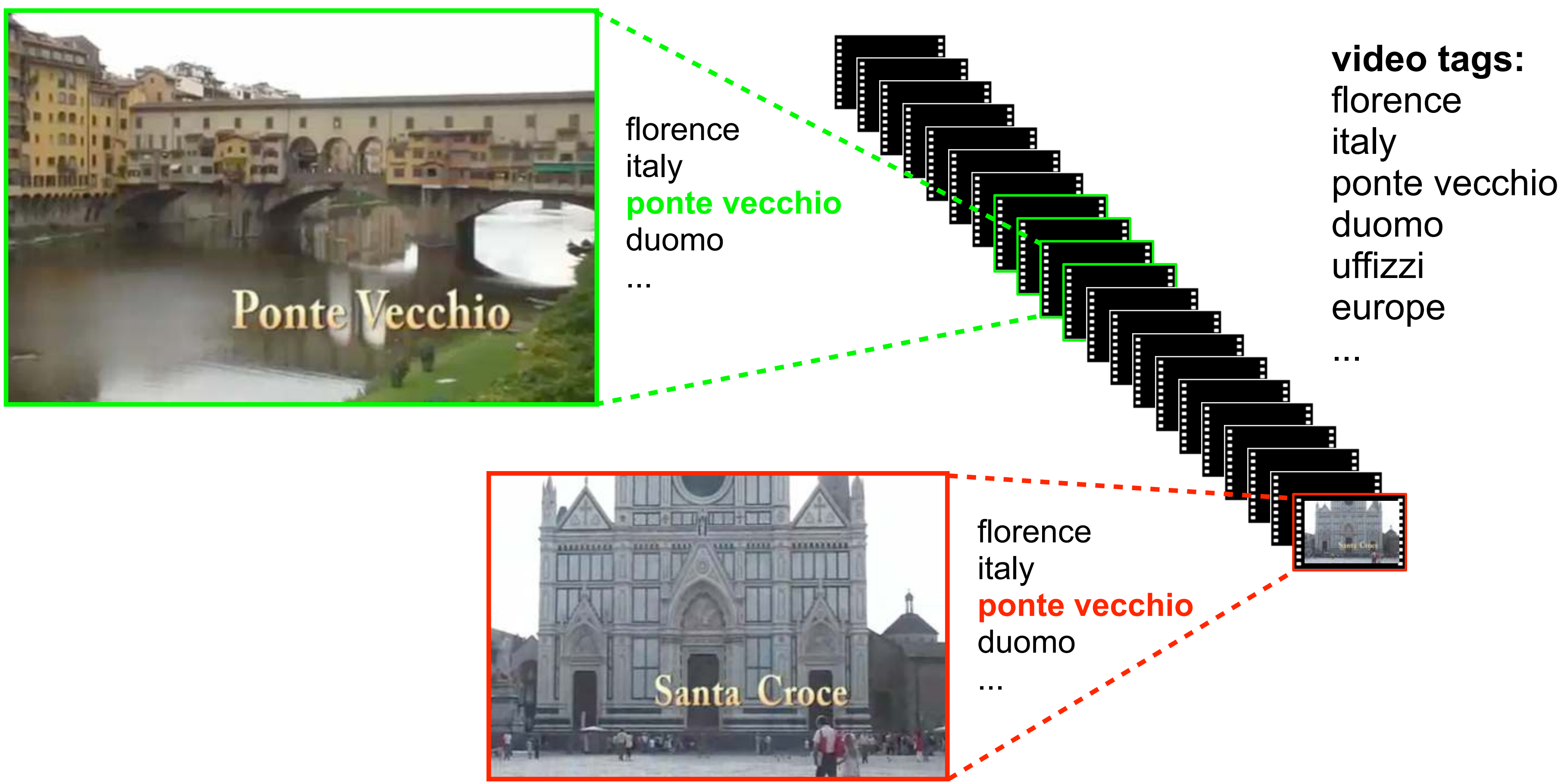


airplane
twin
engine
los
angeles
...

daytime
beach
airplane
ocean
...

- In the case of videos there is also another problem: tags are not “localized” in the video frames

Query tag: ***ponte vecchio***



# Social image retrieval

- **Query-dependent** methods
  - Goal: given a particular query, try to re-rank the results considering the visual content [Hsu *et al.*'07, Jing *et al.*'08]

Query: ***airplane***


- **Query-independent** methods
  - Goal: tag relevance learning by estimating the relevance of each tag with respect to the visual content [Li *et al.*'08 & later, Kennedy *et al.*'09, Wu *et al.*'09]

Query: ***airplane***

**airplane**

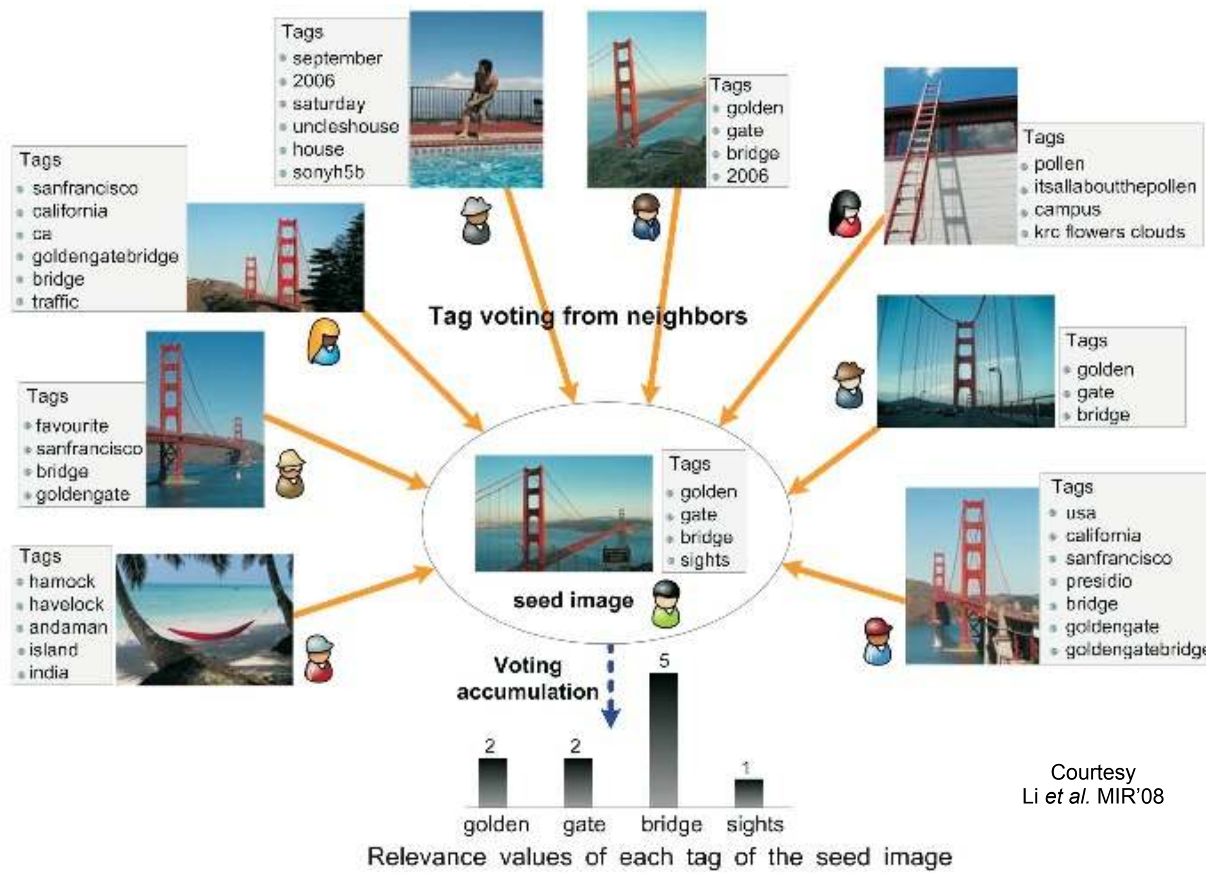
twin
los
angeles

# Tag relevance learning by neighbor voting

- Several recent works focus on the *tag relevance learning* approach since it is more general (i.e. it can be used also as a starting point for query-dependent methods)
- An example: estimate tag relevance by exploiting annotations from neighbors users selected by visual similarity [Li *et al.* ’08,’09]

  – use visual features to describe the content
  – find neighbors by clustering of visual features
  – voting accumulation to learn tag relevance

  – use a multi-feature tag relevance learning to improve results [Li *et al.* 2010]




Courtesy
Li *et al.* MIR’08

# Social video retrieval

- The problem of social video retrieval and tag suggestion in user-generated videos has been less explored
  - several works use YouTube's "related videos" metadata to enrich/re-rank information related to a specific video [Wu *et al.*'09, Liu *et al.*'10]
  - other recent works retrieve visual near-duplicates for tag-suggestion and video re-ranking [Siersdorfer *et al.*'09, Zhao *et al.*'10]

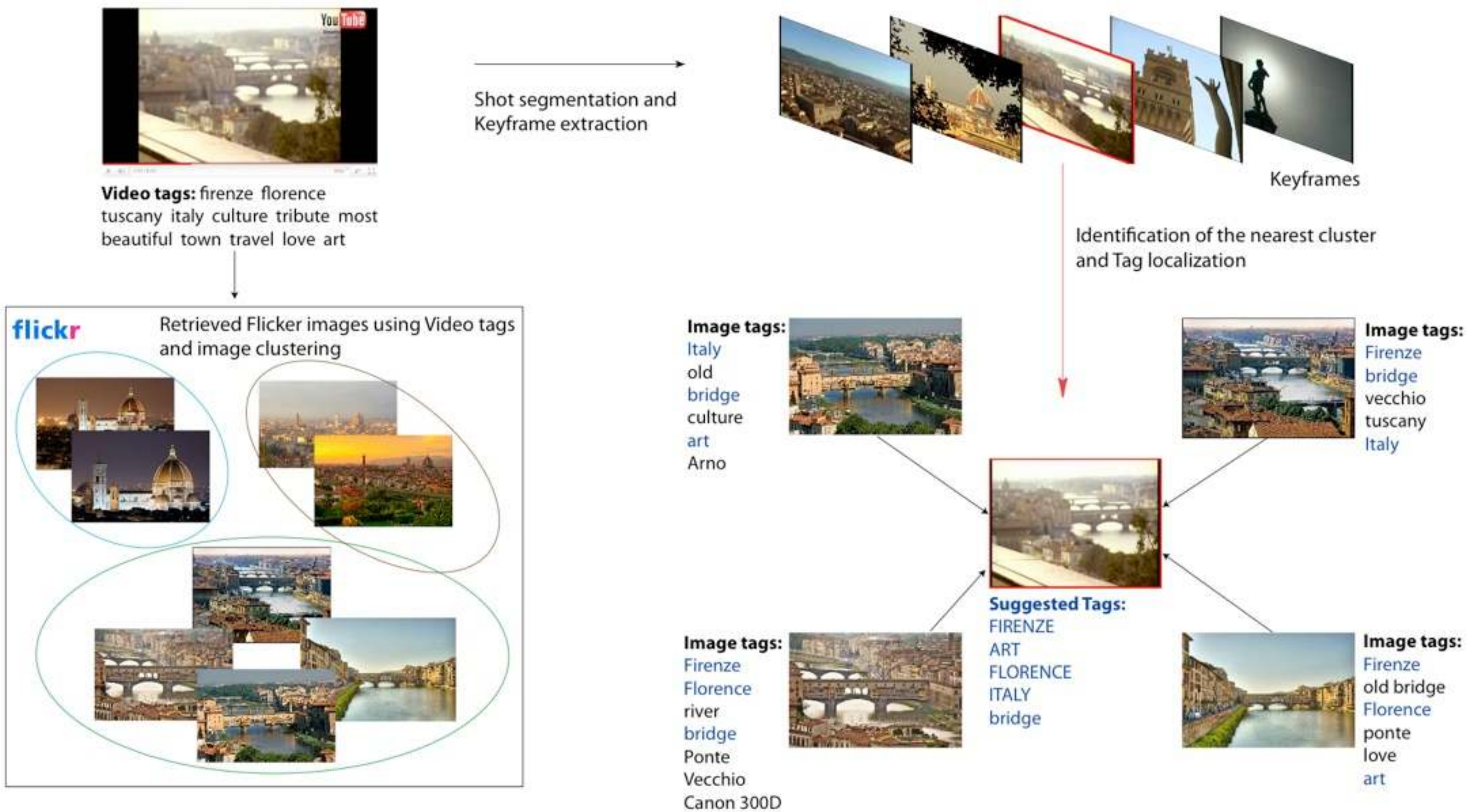
- New tags are usually suggested at the video level
- To the best of our knowledge there are no previous works that try to locate tags within the user-generated video

# Our approach

- We propose an approach for *video tag suggestion* and *temporal localization* based on collective knowledge and visual similarity of video frames

- Our goal is two-fold:
  - exploits tags associated to user-generated videos and images uploaded to social websites (such as YouTube and Flickr) and their visual similarity for tag suggestion at the video level
  - associate the tags to the relevant shots that compose the video

# Overview of the proposed system

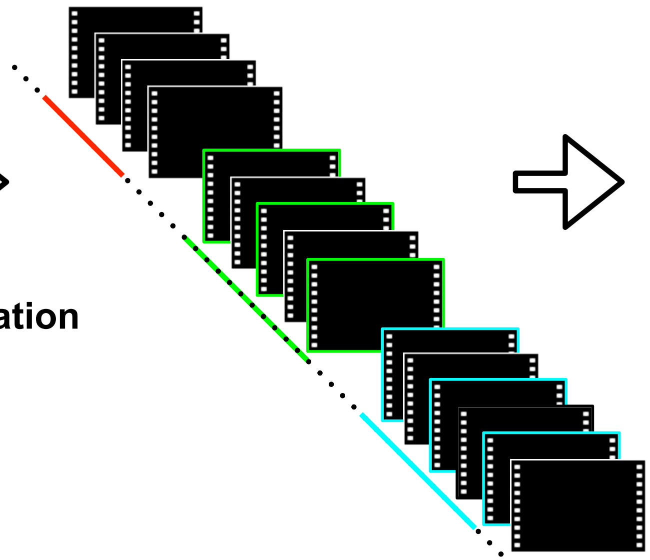


# Exploiting tag relevance for video annotation

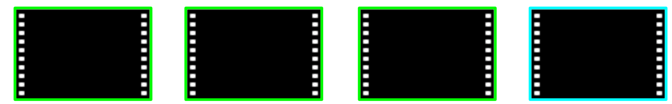
YouTube video


**video tags:**
florence
italy
ponte vecchio
duomo
uffizzi
europe
...

**shot segmentation**



From each shot are extracted 3 keyframes (start, middle, end)



...

$$K = \{k_1, ..., k_o\}$$

$$V = \{v_1, ..., v_n\}$$

The video tags $V$ are used to select and download images from Flickr

flickr


Let $T$ be the union of all the tags of the set of downloaded images $I$

$$I = \{I_{v_1}, ..., I_{v_n}\}$$

$$T = \{t_1, t_2, ..., t_k\}$$

- The set $T$ is considered as the dictionary to be used for the video annotation
- Since it is obtained from social images (Flickr) it is fundamental to evaluate the relevance of the terms in the dictionary
  - to this end we followed and extend the approach of [Li *et al.*'08] to cope with video shot annotation
  - practically tag relevance learning is computed by counting occurrences of each tag $t$ in the *k*NN images, minus the prior frequency of $t$

- For all the keyframes in $K$ and images in $I$ is computed a 72-dim visual feature vector representing global information (color and texture)
  - 48-dim *color correlogram* computed in the HSV color space
  - 6-dim for *color moments* computed in the RGB color space
  - 18-dim for 3 *Tamura features* that account for texture information

$$[d_1, ..., d_{48}, d_{49}, ..., d_{54}, d_{55}, ..., d_{72}]$$

*color correlogram* — *color moments* — *texture (Tamura)*

- Images in $I$ are clustered using k-means and cluster centers are used as an index for ANN-search based on visual similarity to the keyframes in $K$
  - for each keyframes $k$ in $K$ is retrieved the NN cluster center and the images belonging to that clusters are selected as neighbors for $k$
  - tags related to all these images are associated to keyframe $k$, resulting in the tag set $T_k = \{v_1, ..., v_n\}$
  - video tags in $V$ are kept only if they are present in the visual neighborhood (otherwise they are eliminated from the tag list)
  - also the WordNet synonyms of all the tags $v_i$ are used to download images from Flickr (we download only 1/3 of images with respect to the original term)

- To add new tags to each shot we compute a set of candidate tags computed from the dictionary $T$
  - for each $t$ in $T$ is computed its tag relevance and resulting rank position $rank_i$
  - a new tag candidate list $C$ is computed with all the tags $c$ having a co-occurrence value above the average
  - for each $c$ is computed a suggestion score, $score(c,T_k)$, according to the Vote+ algorithm
  - finally, for each candidate tag $c$ of each keyframe $k$, is computed the following suggestion score:

$$score(c,k) = score(c,T_k) \cdot \frac{\lambda}{\lambda + (rank_c - 1)}$$

  - the score is used to order the tags to be added to the shot (only the five most relevant are used)

# Experimental results: dataset

- We evaluate the performance of our approach using a dataset designed to represent the variety of content on YouTube
  - 4 YouTube videos for each YouTube category (1135 shots, 3405 keyframe)
  - all the dataset videos had been previously tagged by YouTube users

- For each YouTube tag our system downloads 15 Flickr images
- In the WordNet query expansion experiment the system downloads 5 additional Flickr images for each WordNet synonym

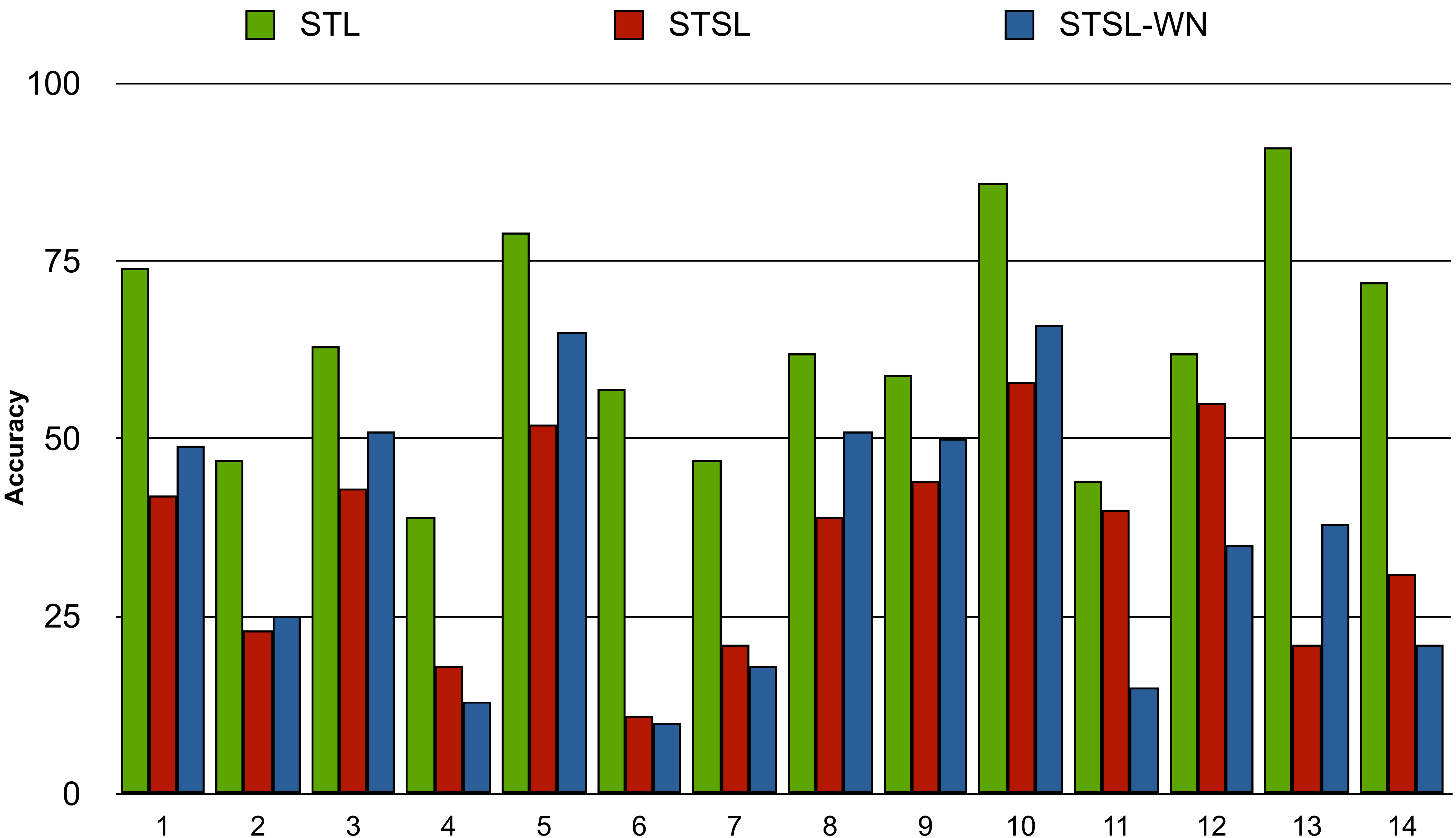
- Output is shown using SRT subtitles
  - Uppercase: original YouTube tags
  - Lowercase: suggested tags for the shot

# Experimental results: types of experiments

- **Shot level Tag Localization (STL)**
  - evaluation of performance of the localization of the user-generated YouTube tags in the correct shots, in terms of accuracy

- **Shot level Tag Suggestion and Localization (STSL)**
  - this measure shows the accuracy of the tag localization at shot level for both user-generated and suggested tags

- **STSL with WordNet query expansion (STSL-WN)**
  - accuracy of STSL with WordNet synset expansion of the YouTube tags that have been kept at the end of localization process




1. Cars & Vehicles
2. Comedy
3. Education
4. Entertainment
5. Film & Animation
6. Gaming
7. Howto & Style
8. Music
9. News & Politics
10. People & Blogs
11. Pets & Animals
12. Science & Technology
13. Sport
14. Travel & Events

Scene 14: PARK, TERRAIN, LAND, landscape, sky, mountain, scenery, colors


Scene 1: VOLCANO, ERUPTION,
EYJAFJALLAJÖKULL, ICELAND,
glacier, landscape,
volcaniceruption, eldgos, nature


Scene 1: MAID, MIST, NIAGRA, FALLS,
scotland, waterfall, trees,
crossdresser, tablier

Thank You