

Visual words and near duplicate detection

D.A. Forsyth

Near duplicate detection

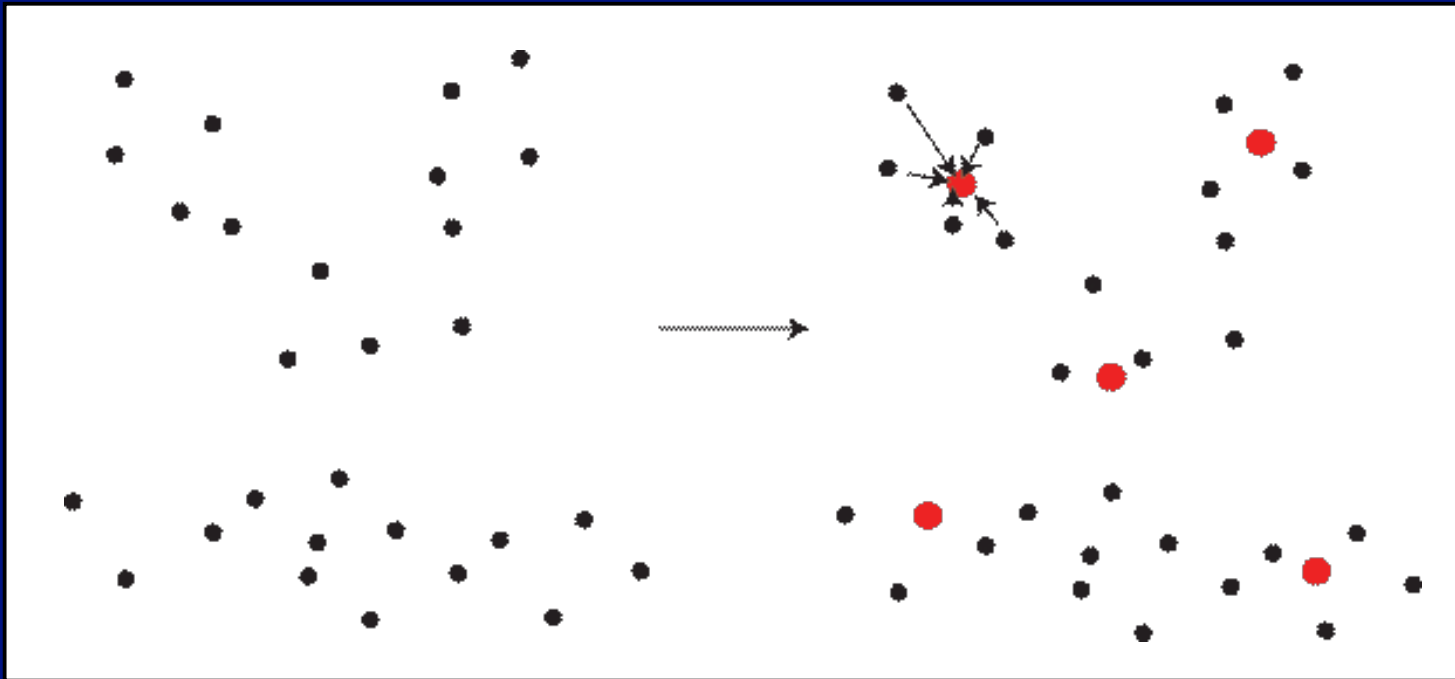
- Applications
 - Trademarks
 - Copyright
 - Spam filtering
 - Filtering search results
 - Video search

Simple information retrieval

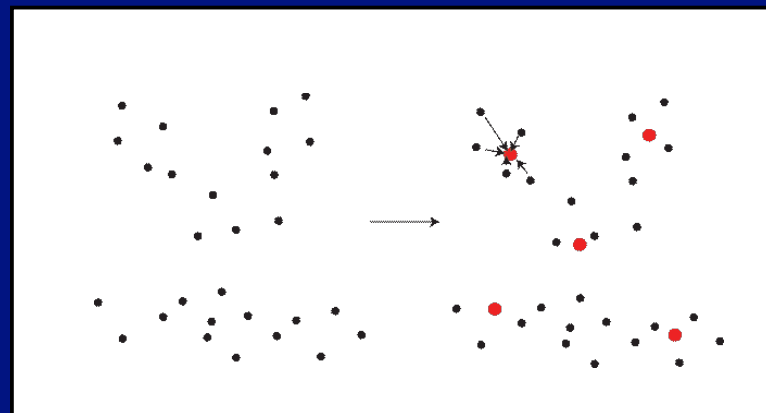
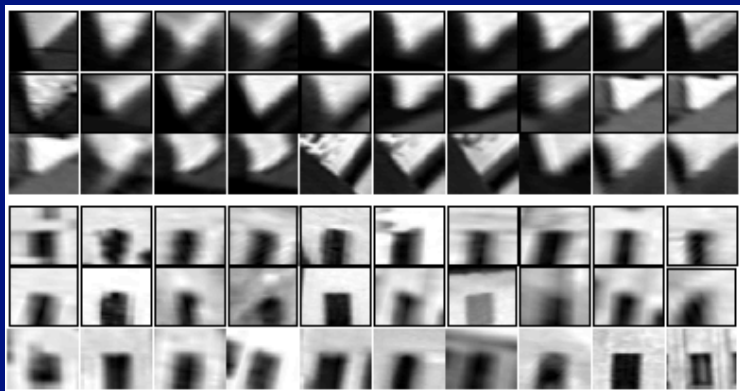
- For documents
- Model documents as bags of words
- Fact:
 - most words are rare
 - word frequency/occurrence is revealing
- Similarity
 - cosine distance between word frequency vectors
 - 0-1
 - frequency
 - TF-IDF weighted
- Find documents with inverted index

Idea: visual words

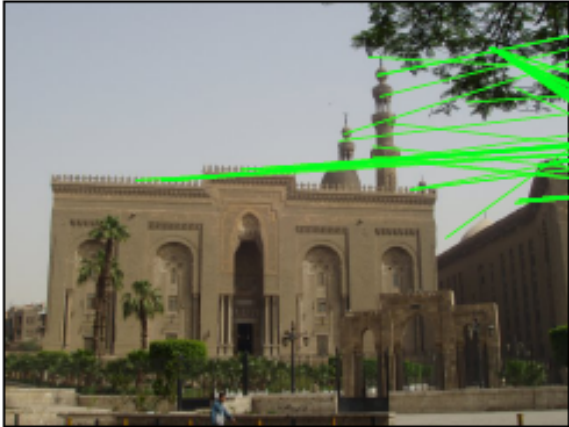
- Turn patches into “words”
 - then we can do all above, easily
- Do this with vector quantization



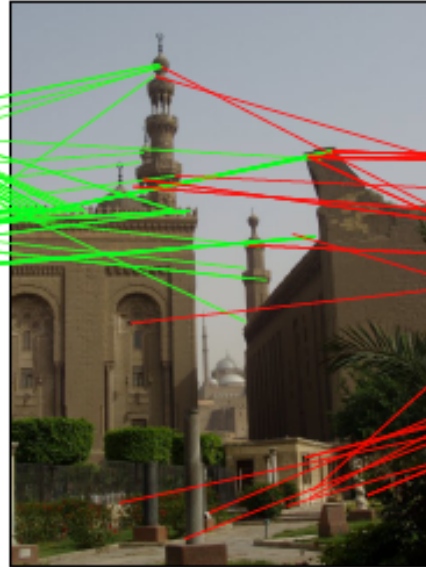
Visual words



69 matches



(b)



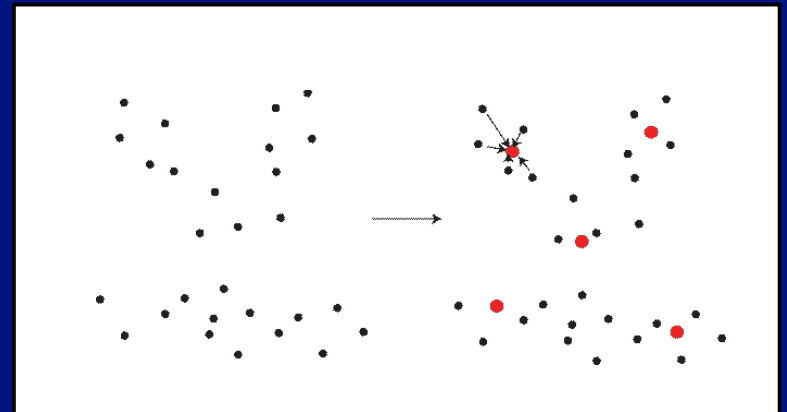
35 matches



Jegou et al, 10

Visual words

- Choose a vocabulary
 - Cluster centers
 - choose a fixed number of cluster centers
 - find centers using k-means clustering of vectors describing patches
- Test image
 - take patch
 - word is closest cluster center



K-Means

- Choose a fixed number of clusters
- Choose cluster centers and point-cluster allocations to minimize error

$$\sum_{i \in \text{clusters}} \left\{ \sum_{j \in \text{elements of } i\text{'th cluster}} \|x_j - \mu_i\|^2 \right\}$$

- can't do this by search
 - there are too many possible allocations.
- Algorithm
 - fix cluster centers; allocate points to closest cluster
 - fix allocation; compute best cluster centers
 - x could be any set of features for which we can compute a distance (careful about scaling)

Finding near duplicate images - I

- Set up dataset
 - For many images
 - find all interest points
 - compute HOG descriptors
 - Use k-means to compute cluster centers
 - For each image
 - map each interest point to closest cluster center
 - make histogram of cluster centers
 - insert histogram vectors into approximate nearest-neighbor datastructure

Finding near duplicate images - II

- Query with image
 - For query image
 - Find all interest points
 - compute HOG descriptors
 - map to cluster centers
 - form count vector
 - Query ANN datastructure with count vector
 - Rank returned images by
 - distance
 - cosine distance
 - other functions
 - Report the best

Finding near duplicate images - III

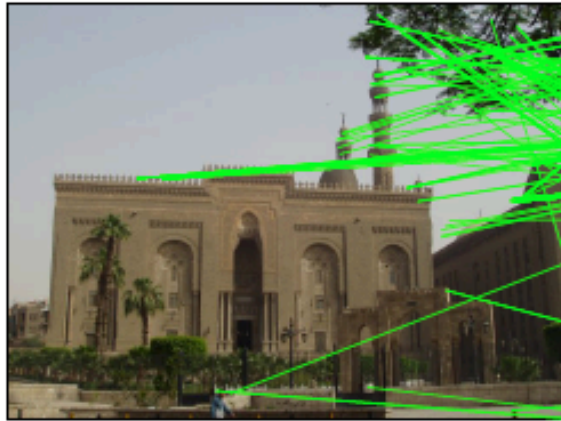
- Important variants
 - TF-IDF weighting
 - More/fewer visual words
 - Local consistency checks
 - Video NDD
 - temporal as well as spatial
 - eg paths of words
 - Object query
 - Adjust distance function

Object query

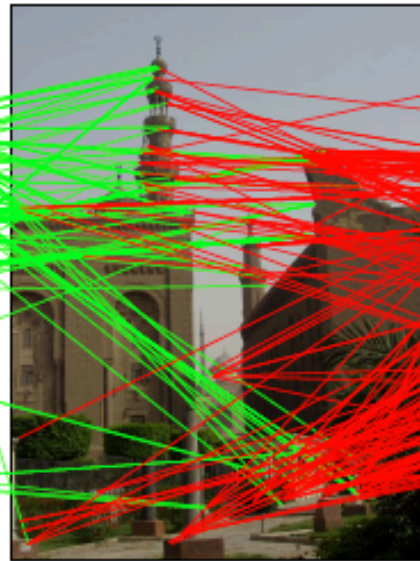


Sivic and Zisserman 03

201 matches



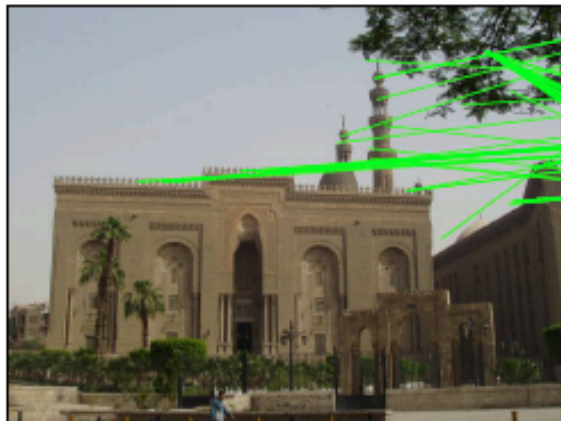
(a)



240 matches



69 matches



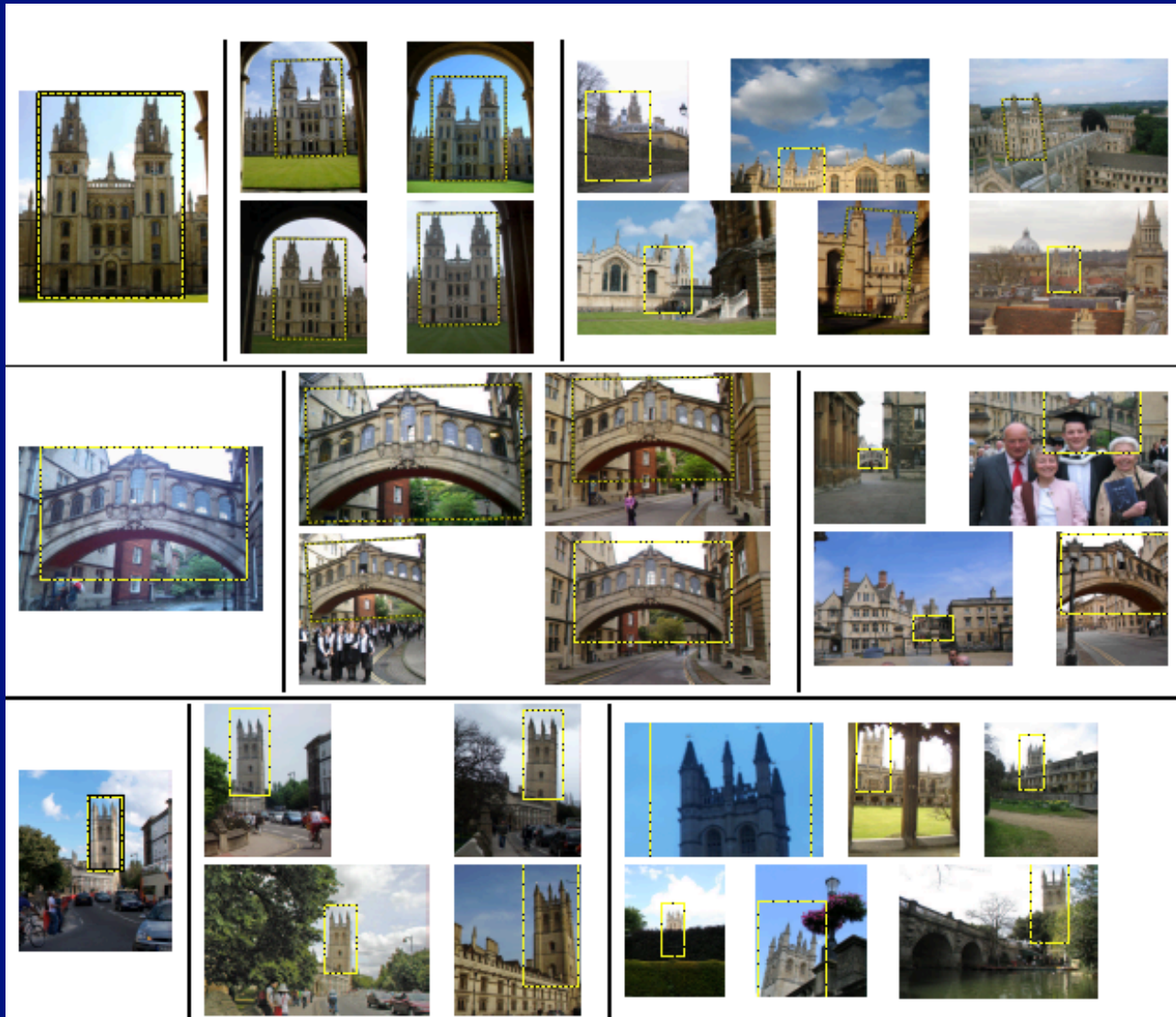
(b)



35 matches

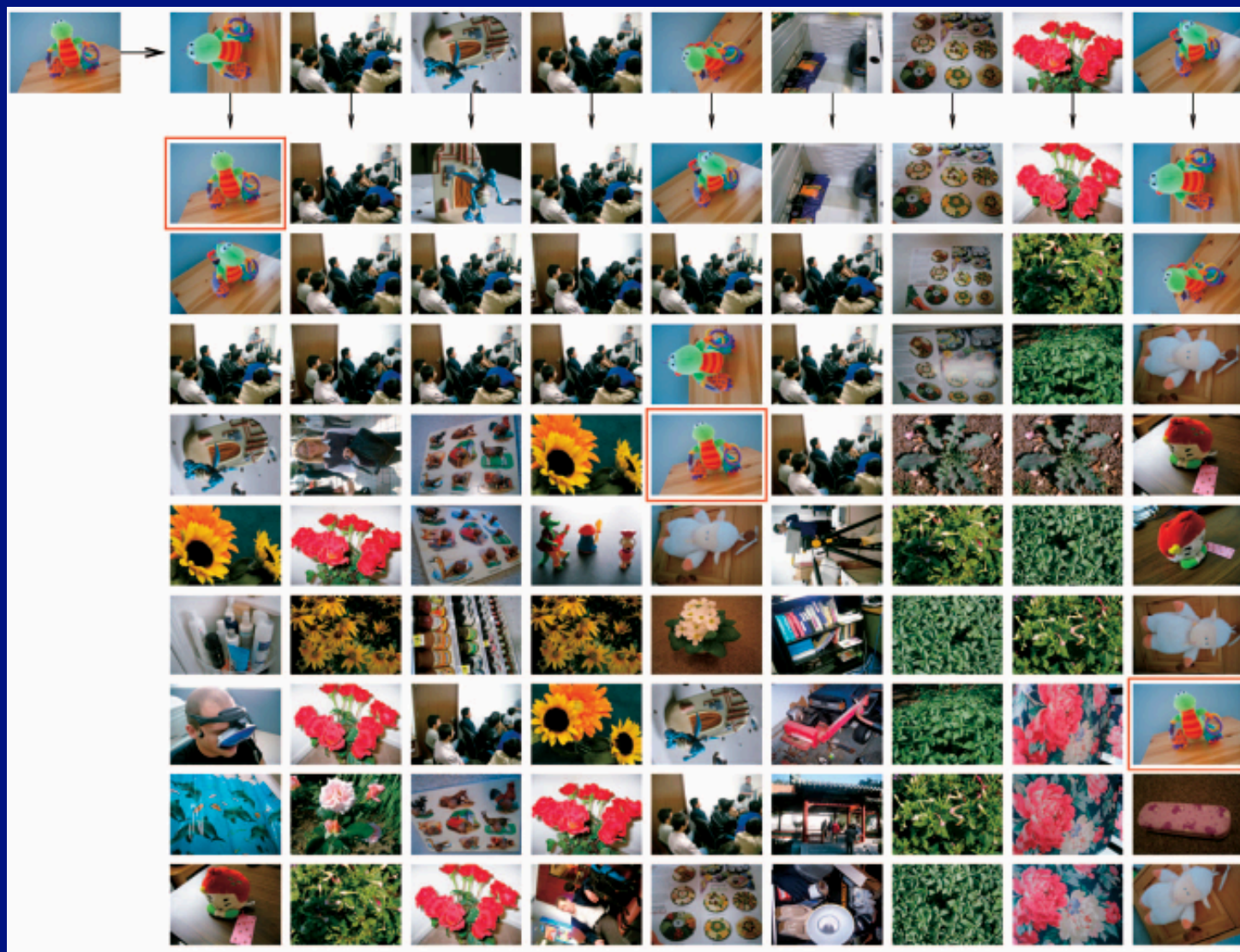


Query expansion



Chum et al 07

Adjusting distance



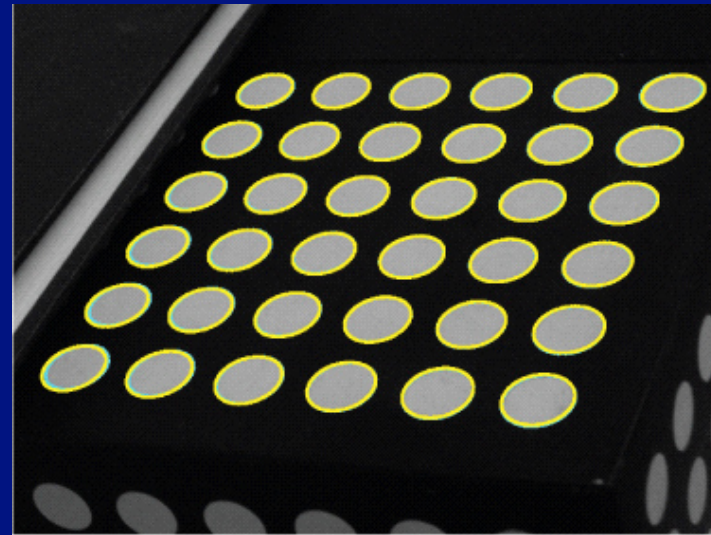
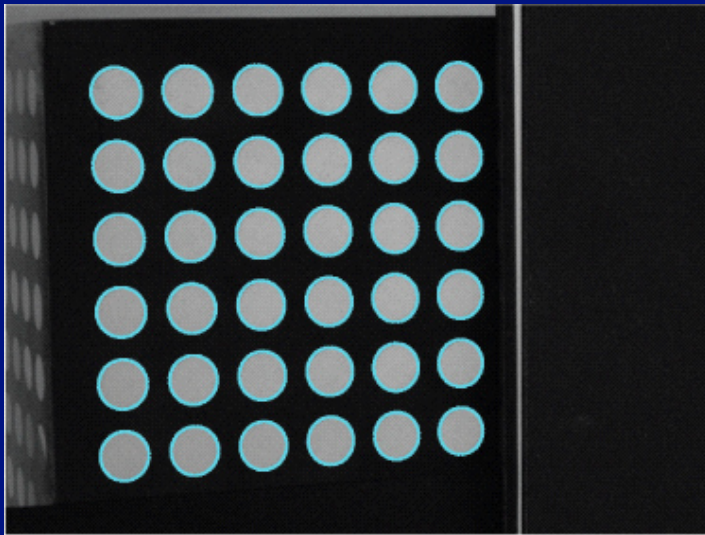
Jegou et al 10

Transformation issues

- Our model doesn't completely cover all deformations
 - eg view a patch at an odd angle
 - more complex transformation than scaling
- It is useful to model plane to image transformations



Homographies



- Equation:
 - (u, v) coordinates on object plane
 - (x, y) coordinates on image plane

$$x = \frac{a_{00}u + a_{01}v + a_{02}}{a_{20}u + a_{21}v + a_{22}}$$

$$y = \frac{a_{10}u + a_{11}v + a_{12}}{a_{20}u + a_{21}v + a_{22}}$$