Unsupervised Joint Object Discovery and Segmentation in Internet Images

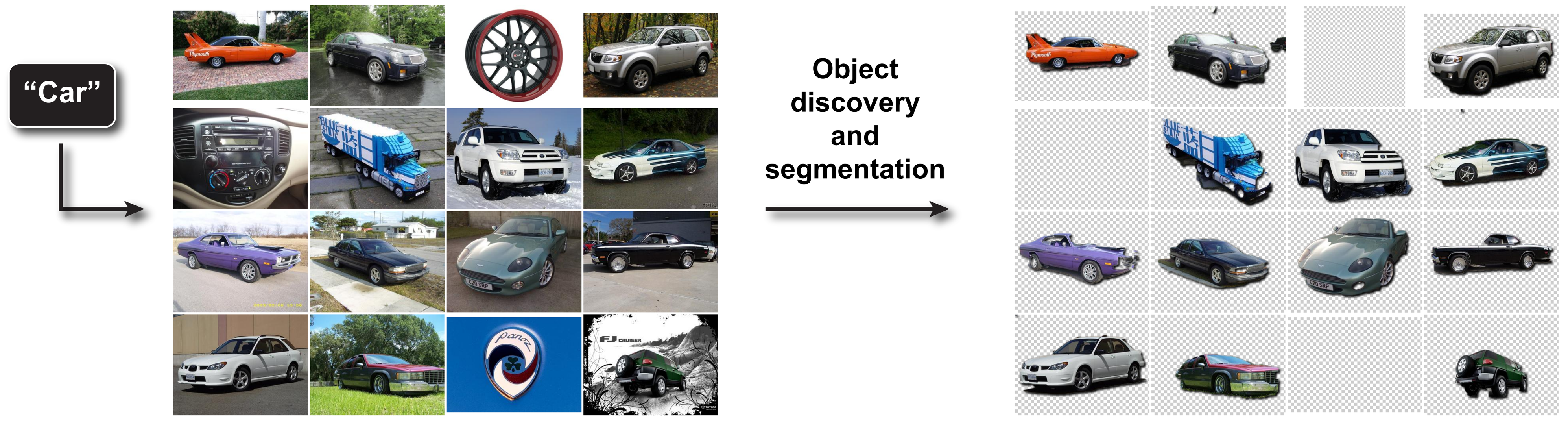
Michael Rubinstein[1,3] Armand Joulin[2,3] Johannes Kopf[3] Ce Liu[3]

[1]MIT CSAIL [2]INRIA [3]Microsoft Research

Object Discovery and Segmentation

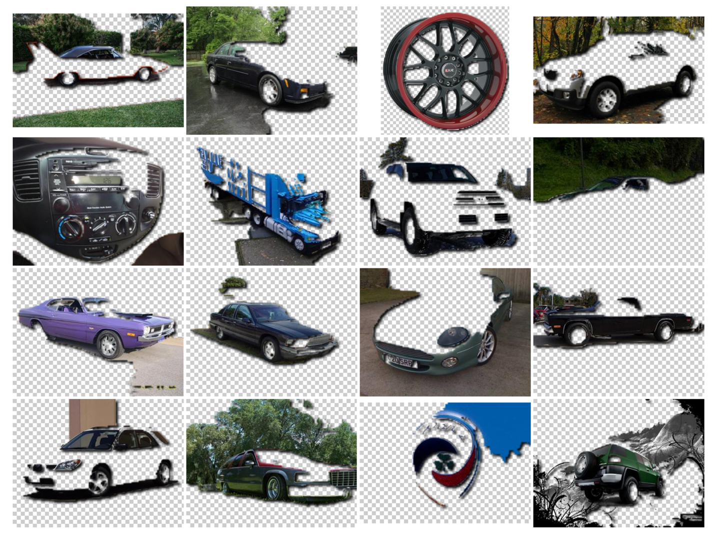
Goal: Automatically segment out the common object in a set of images, without additional info on the images or the object

Challenges in Internet datasets: (a) Extreme variation in appearance (color, texture, shape, pose, size, location, ...)
(b) Many ***noise*** images (image not containing the common object)



Our automatic segmentation results

This paper: An unsupervised algorithm that can segment the common visual category(ies) in ***noisy*** image datasets. Performs considerably better than previous co-segmentation methods on Internet datasets.



Recent co-segmentation results [Joulin12]

Basic Idea: Jointly discover and segment the object in all the images!

Pixels (features) belonging to the common object should be:

(a) **Salient** - dissimilar to other pixels (features) in their image } Captured by image ***saliency measures***

(b) **Sparse** - similar to other pixels (features) in other images } Captured by (dense) **image correspondence**

(with respect to smooth transformations between the images)

Input: image dataset $\boldsymbol{I} = \{I_1, \dots I_N\}$

Output: binary masks $\boldsymbol{B} = \{\boldsymbol{b}_1, \dots \boldsymbol{b}_N\}$, $\boldsymbol{b}_i(\boldsymbol{x}) = 1 \rightarrow$ Foreground (the common object)
$\boldsymbol{b}_i(\boldsymbol{x}) = 0 \rightarrow$ Background (not the object)

Foreground likelihood (data term):

$$\Phi^i(\boldsymbol{x}) = \begin{cases} \Phi^i_{saliency}(\boldsymbol{x}) + \lambda_{match}\Phi^i_{match}(\boldsymbol{x}), & \boldsymbol{b}_i(\boldsymbol{x}) = 1 \\ \beta, & \boldsymbol{b}_i(\boldsymbol{x}) = 0 \end{cases}$$

$$\Phi^i_{saliency}(\boldsymbol{x}) = -\log M_i(\boldsymbol{x})$$

M_i - Saliency maps computed with an off-the-shelf measure [Cheng11]
(dataset-wide normalized)

$$\Phi^i_{match}(\boldsymbol{x}) = \frac{1}{N_i}\sum_{j \in N_i}\left\|S_i(\boldsymbol{x}) - S_j(\boldsymbol{x} + \boldsymbol{w}_{ij}(\boldsymbol{x}))\right\|_1$$

S_i - Dense SIFT descriptors of image I_i
$\boldsymbol{w}_{ij}$ - Pixel correspondence $I_i \rightarrow I_j$
N_i - Nearest neighbors of image I_i
(dataset-wide normalized)

Objective function:

$$E(\boldsymbol{B};\boldsymbol{W},\boldsymbol{H}) = \sum_{i=1}^{N}\sum_{\boldsymbol{x}\in\Lambda_i}\Big(\underbrace{\Phi^i(\boldsymbol{x})}_{\text{Likelihood}} + \underbrace{\lambda_{color}\Phi^i_{color}(\boldsymbol{x},\boldsymbol{h}_i)}_{\text{Per-image F/B color model}} + \underbrace{\sum_{\boldsymbol{y}\in N^i_{\boldsymbol{x}}}\lambda_{int}\Psi^i_{int}(\boldsymbol{x},\boldsymbol{y})}_{\text{Intra-image regularization}} + \underbrace{\sum_{j\in N_i}\lambda_{ext}\Psi^{ij}_{ext}(\boldsymbol{x},\boldsymbol{x}+\boldsymbol{w}_{ij}(\boldsymbol{x}))}_{\text{Inter-image regularization}}\Big)$$

$$\Phi^i_{color}(\boldsymbol{x},\boldsymbol{h}_i) = -\log \underbrace{\boldsymbol{h}_i^{\boldsymbol{b}_i(\boldsymbol{x})}}_{\text{3D histograms in color space}}(\boldsymbol{x})$$

$$\Psi^i_{int}(\boldsymbol{x},\boldsymbol{y}) = [\boldsymbol{b}_i(\boldsymbol{x}) \neq \boldsymbol{b}_i(\boldsymbol{y})]\exp(-\|I_i(\boldsymbol{x}) - I_i(\boldsymbol{y})\|_2^2)$$

$$\Psi^{ij}_{ext}(\boldsymbol{x},\boldsymbol{z}) = [\boldsymbol{b}_i(\boldsymbol{x}) \neq \boldsymbol{b}_j(\boldsymbol{z})]\exp\left(-\left\|S_i(\boldsymbol{x}) - S_j(\boldsymbol{z})\right\|_1\right)$$

$$\boldsymbol{z} = \boldsymbol{x} + \boldsymbol{w}_{ij}(\boldsymbol{x})$$

Large-scale graphical model connecting similar images and corresponding pixels

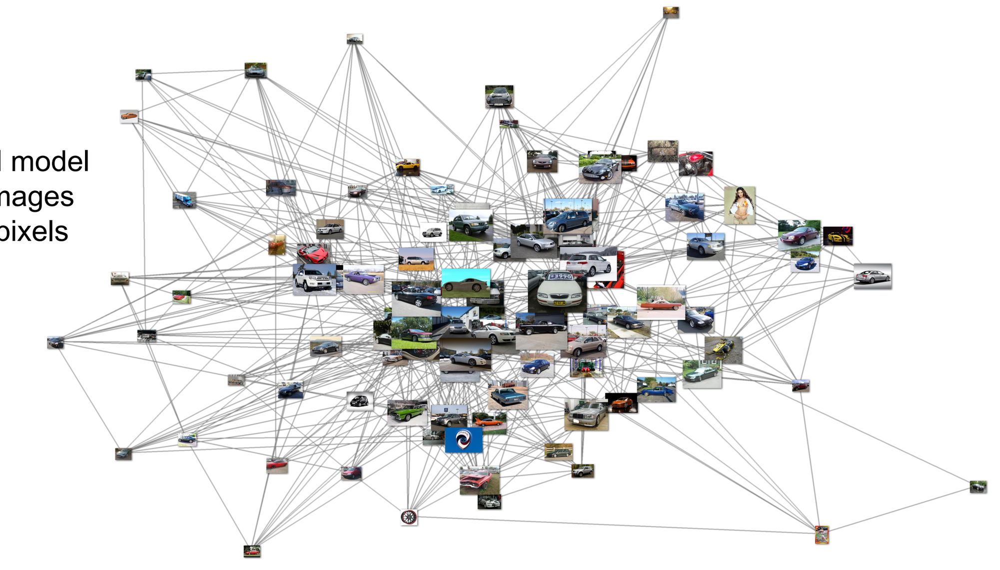
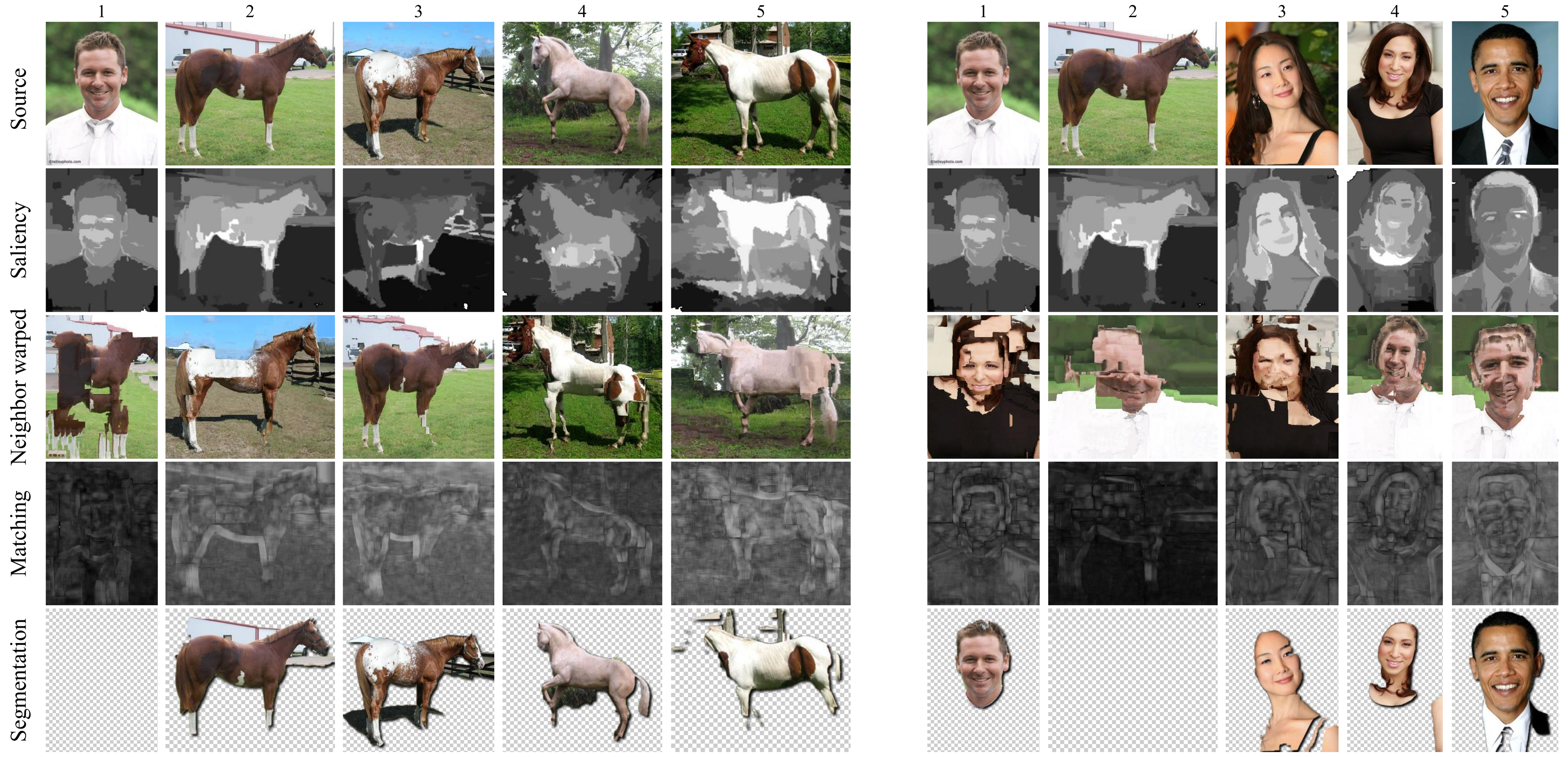


Image correspondence:

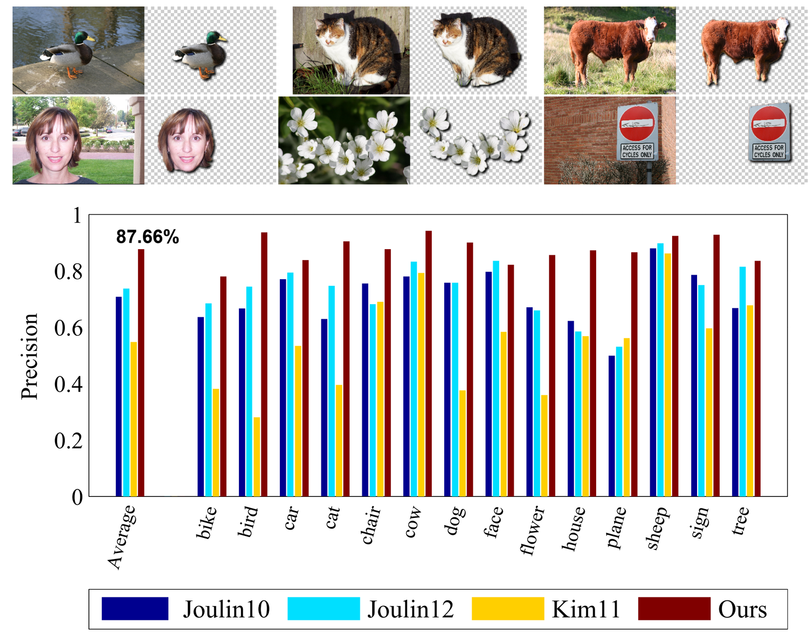
$$E(\boldsymbol{w}_{ij};\boldsymbol{b}_i,\boldsymbol{b}_j) = \sum_{\boldsymbol{x}\in\Lambda_i}\boldsymbol{b}_i(\boldsymbol{x})\Big(\boldsymbol{b}_j(\boldsymbol{x}+\boldsymbol{w}_{ij}(\boldsymbol{x}))\left\|S_i(\boldsymbol{x}) - S_j(\boldsymbol{x}+\boldsymbol{w}_{ij}(\boldsymbol{x}))\right\|_1 + \left(1 - \boldsymbol{b}_j(\boldsymbol{x}+\boldsymbol{w}_{ij}(\boldsymbol{x}))\right)C_0 + \sum_{\boldsymbol{y}\in N^i_{\boldsymbol{x}}}\alpha\left\|\boldsymbol{w}_{ij}(\boldsymbol{x}) - \boldsymbol{w}_{ij}(\boldsymbol{y})\right\|_2\Big)$$

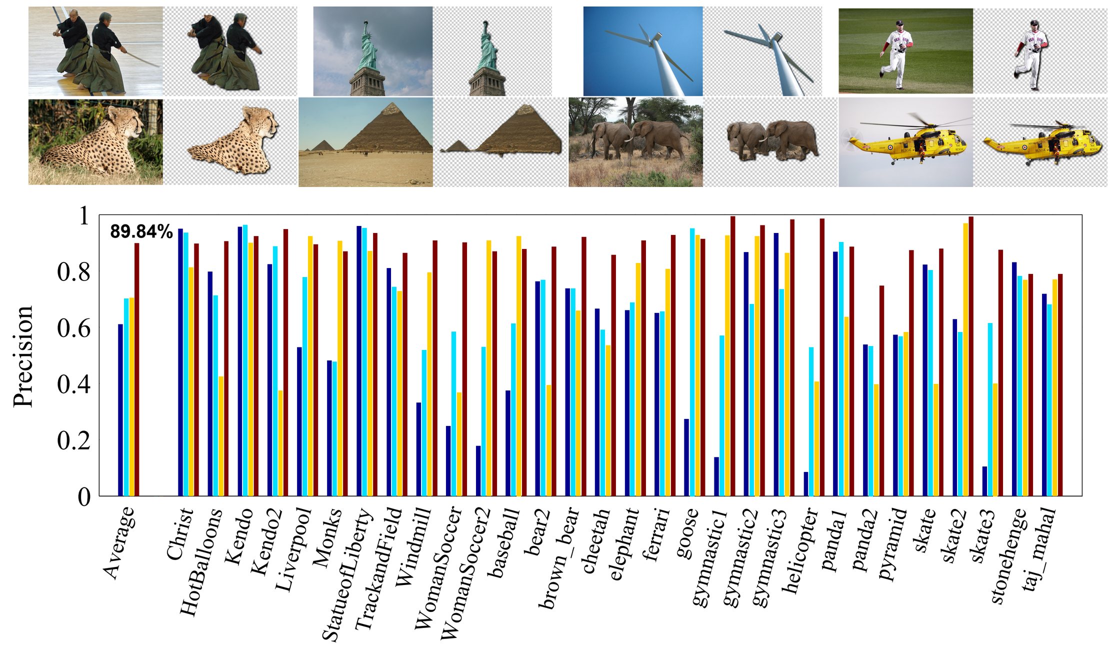
A Small Test Case



Results on Standard Co-segmentation Datasets

- Standard co-segmentation datasets are too simple!
- Can get good (state-of-the-art) accuracy without co-segmentation: $\lambda_{match} = 0, \lambda_{ext} = 0$



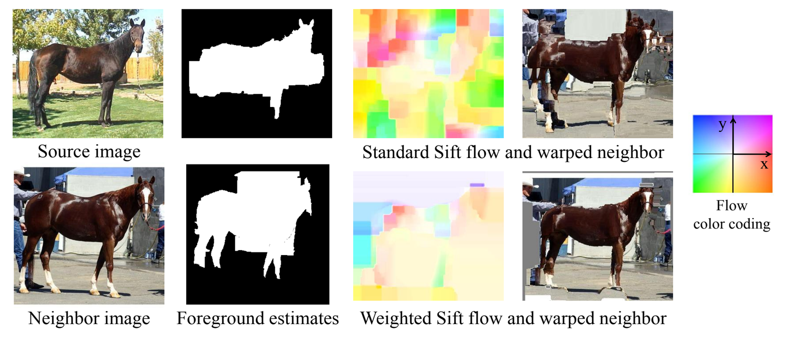


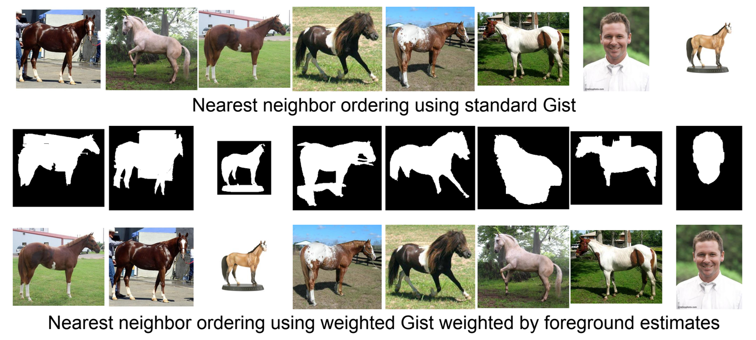
[Joulin10] A. Joulin, F. Bach, and J. Ponce. Discriminative clustering for image co-segmentation. CVPR, 2010
[Joulin12] A. Joulin, F. Bach, and J. Ponce. Multi-class cosegmentation. CVPR, 2012
[Kim11] G. Kim, E. Xing, L. Fei-Fei, and T. Kanade. Distributed cosegmentation via submodular optimization on anisotropic diffusion. ICCV, 2011
[Vicente11] S. Vicente, C. Rother, and V. Kolmogorov. Object cosegmentation. CVPR, 2011
[Cheng11] M. Cheng, G. Zhang, N. Mitra, X. Huang, and S. Hu. Global contrast based salient region detection. CVPR, 2011

Comparison with "Object Cosegmentation":

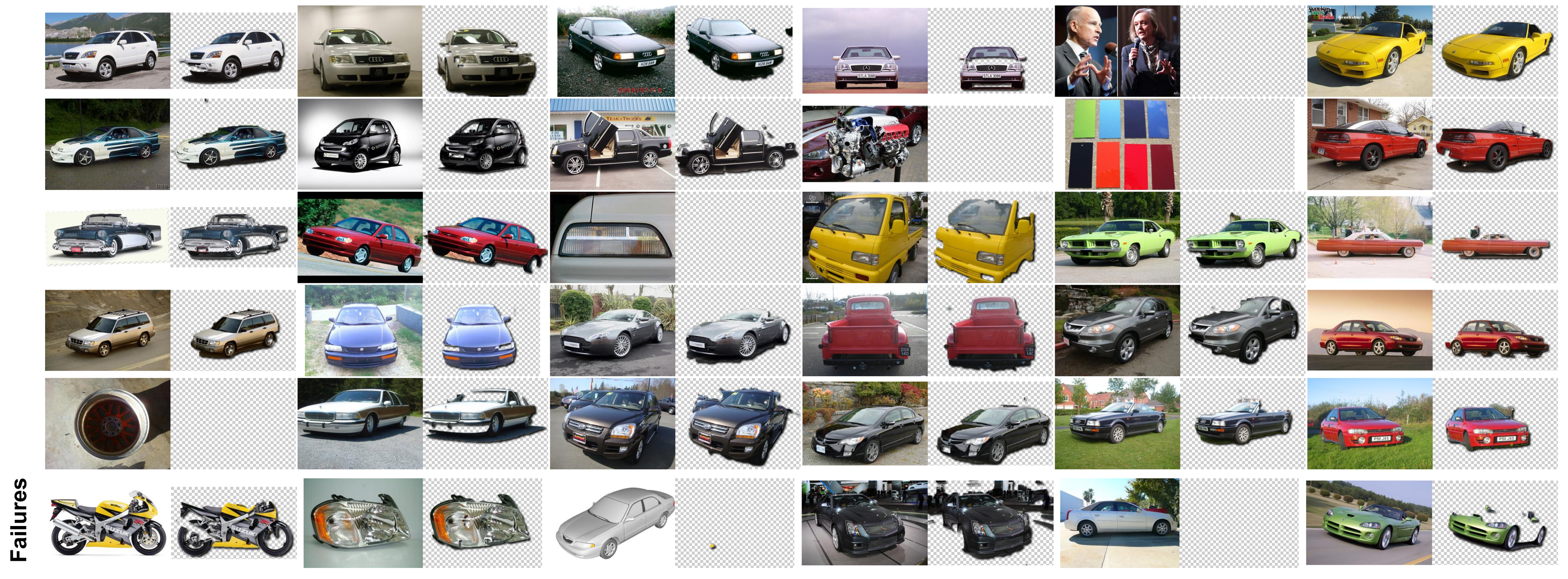
Method	MSRC		iCoseg	
	$\bar{P}$	$\bar{J}$	$\bar{P}$	$\bar{J}$
[Vicente11]	90.2	70.6	85.34	62.04
Ours	**92.16**	**74.7**	**89.6**	**67.63**

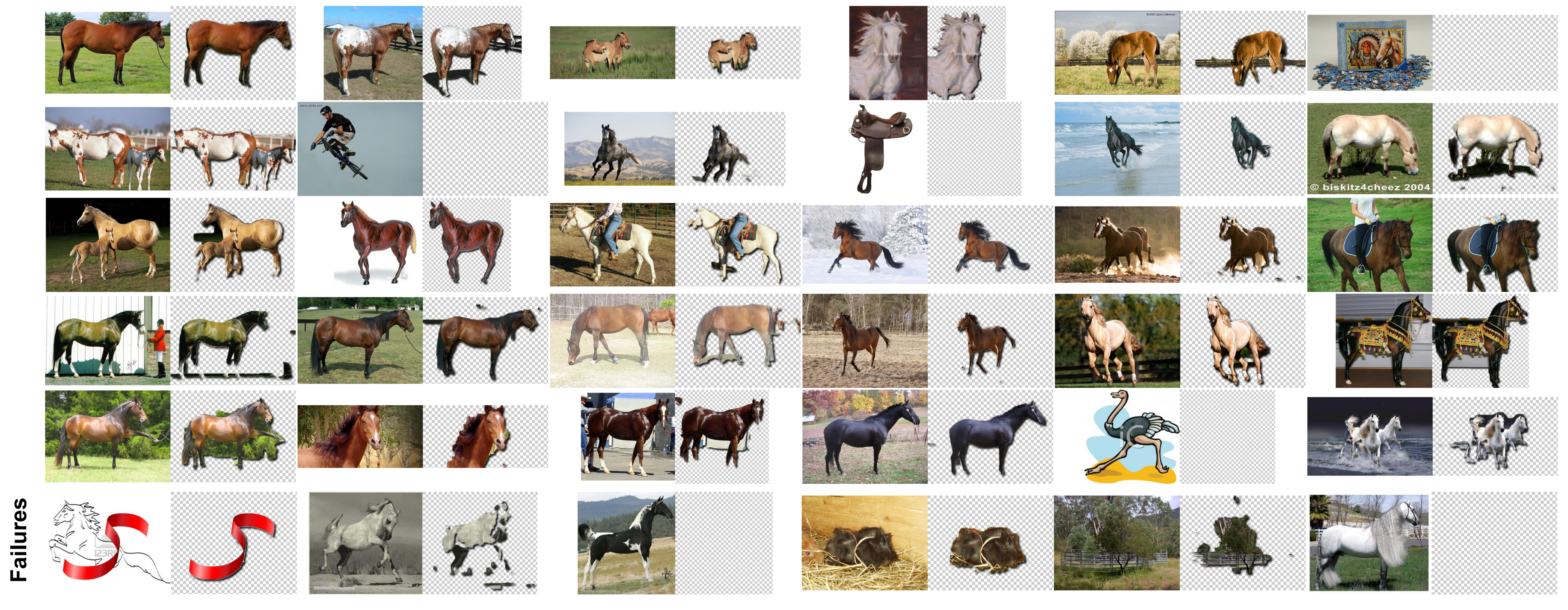
Image Correspondence and Nearest Neighbors





Results on Internet Datasets





Noise level (% images not containing the common object)

Quantitative results with and without utilizing image correspondences:

Method	Car (7.5%)		Horse (7.8%)		Airplane (16%)	
	P	J	P	J	P	J
Without corr.	72.25	46.10	74.88	50.06	80.53	51.18
With corr.	**83.38**	**63.36**	**83.69**	**53.89**	**86.14**	**55.62**
	4,347 images		6,381 images		4,542 images	

Comparison with recent co-seg methods (100 randomly selected images from each dataset):

Method	Car (11%)		Horse (7%)		Airplane (18%)	
	P	J	P	J	P	J
Baseline 1	68.91	0	81.54	0	87.48	0
Baseline 2	31.09	34.93	18.46	19.85	12.52	15.26
[Joulin10]	58.7	37.15	63.84	30.16	49.25	15.36
[Joulin12]	59.2	35.15	64.22	29.53	47.48	11.72
[Kim11]	68.85	0.04	75.12	6.43	80.2	7.9
Ours	**85.38**	**64.42**	**82.81**	**51.65**	**88.04**	**55.81**