A generic model to compose vision modules for holistic scene understanding

Adarsh Kowdle*, Congcong Li*,
Ashutosh Saxena, and Tsuhan Chen

Cornell University, Ithaca, NY, USA

*indicates equal contribution

Outline

- Motivation
- Model
- Algorithm
- Results and Discussions
- Conclusions

Motivation

Motivation

Scene Understanding

Object Detection

Depth Estimation

Event Categorization

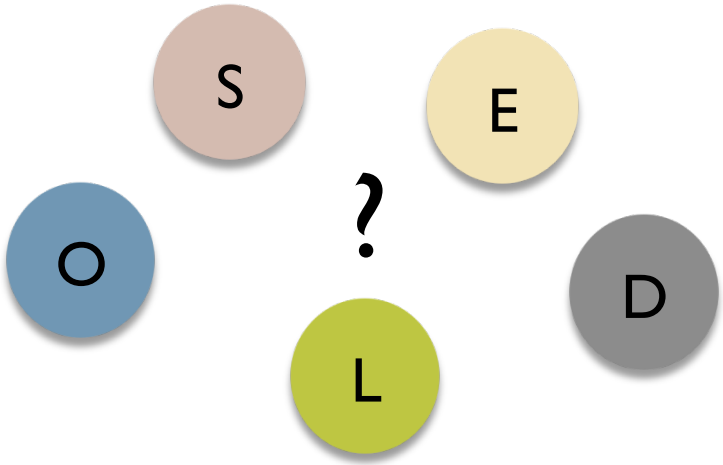
Scene Categorization

Saliency Detection

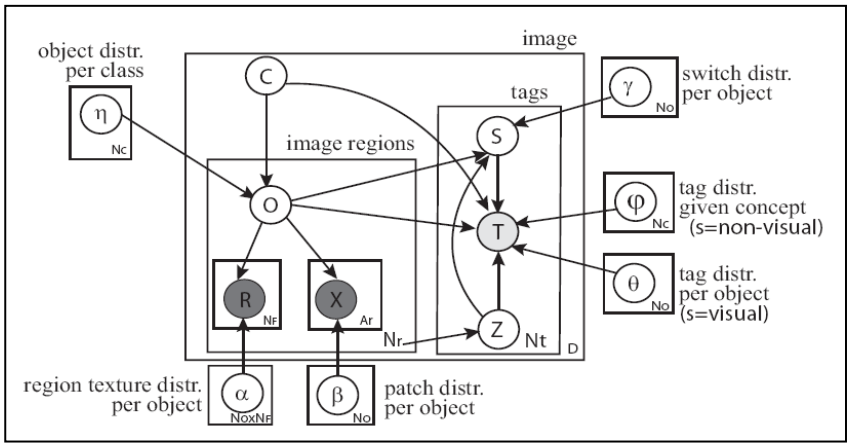
Geometric Layout

. . .

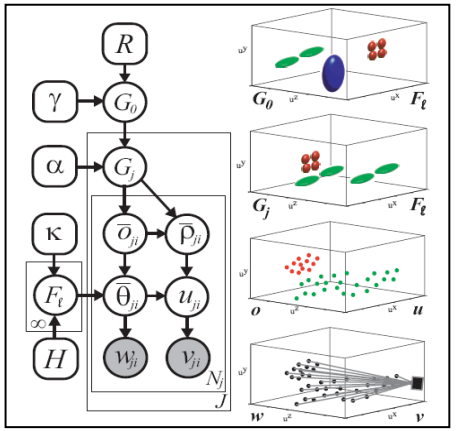
Vision tasks are highly related.
But, how do we connect them?



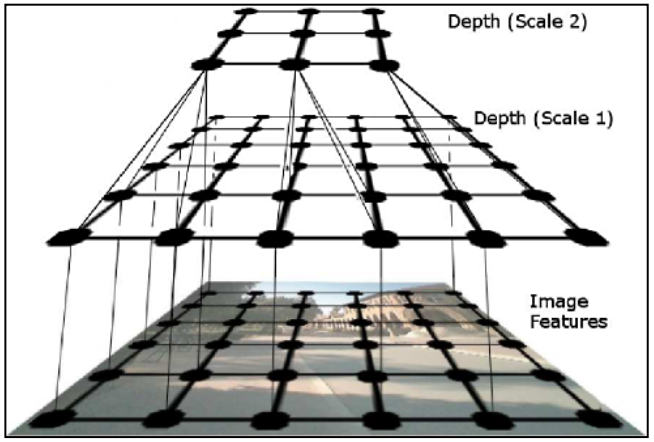
Motivation



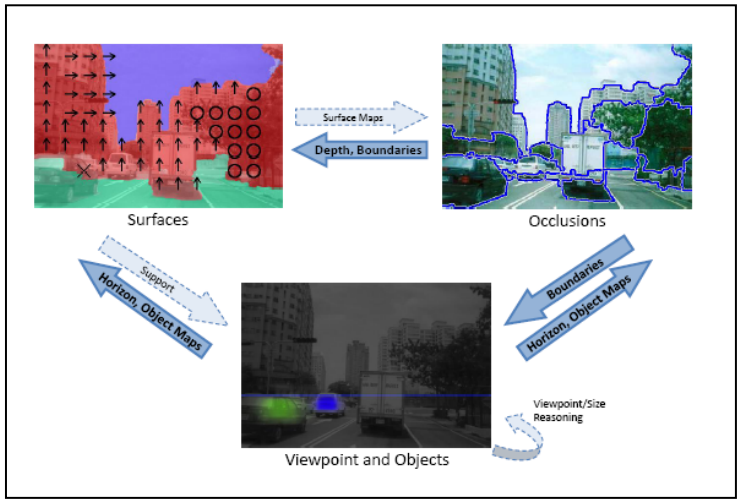
Li et al, CVPR'09



Sudderth et al, CVPR'06

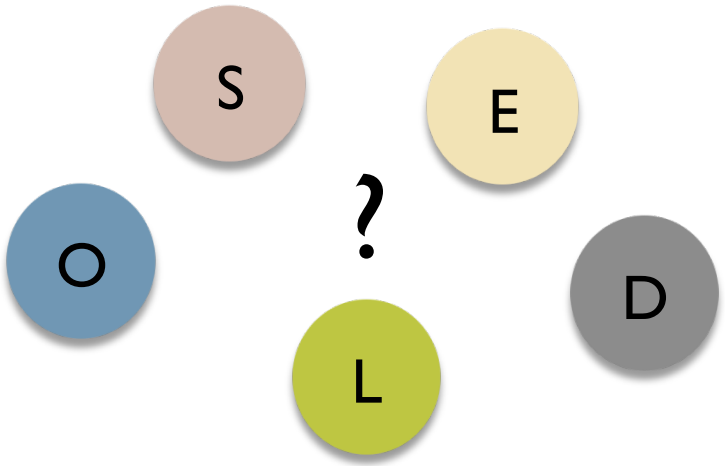


Saxena et al, IJCV'07



Hoiem et al, CVPR'08

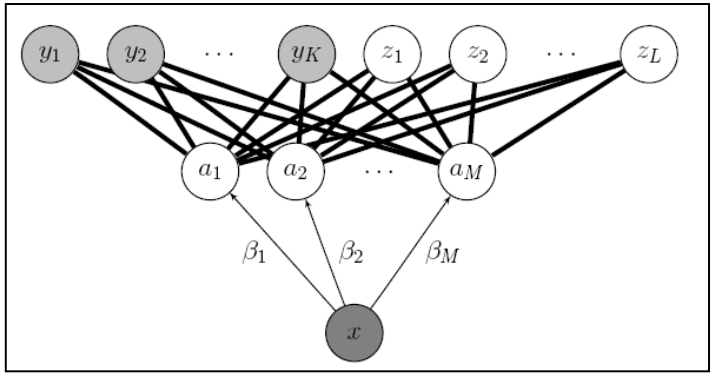
Motivation



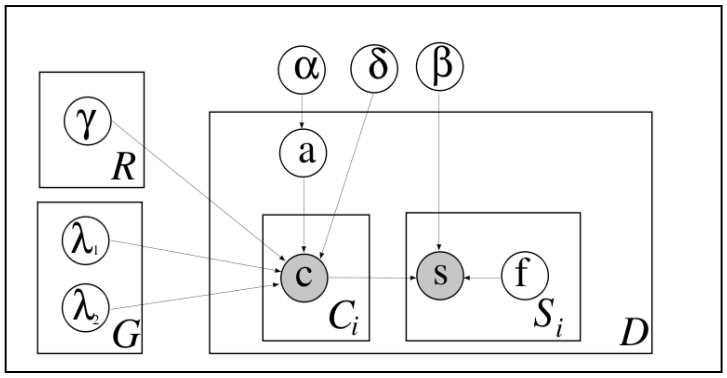
- A generic model which can treat each classifier as a "black-box" and compose them to incorporate the additional information automatically

Motivation

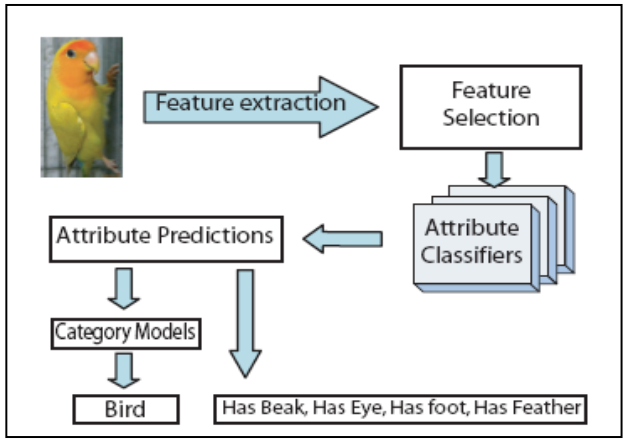
Visual attributes



Lampert et al, CVPR'09



Ferrari et al, NIPS'07



Farhadi et al, CVPR'09

Wang et al, ICCV'09

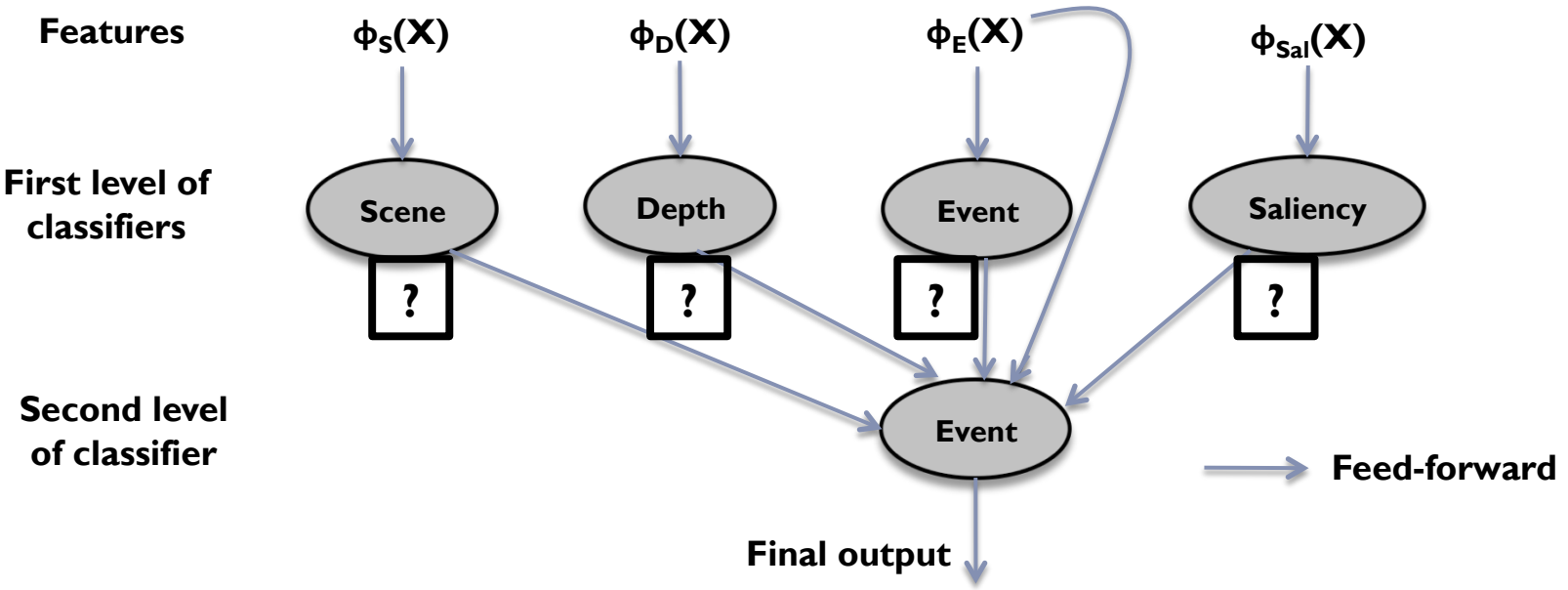
Motivation

- Attributes for scene understanding?

Bocce
"opencountry-like scene" attribute
"salient region" attribute
"depth in the middle region" attribute

- A model which can compose the "black-box" classifiers and automatically exploit attributes for scene understanding

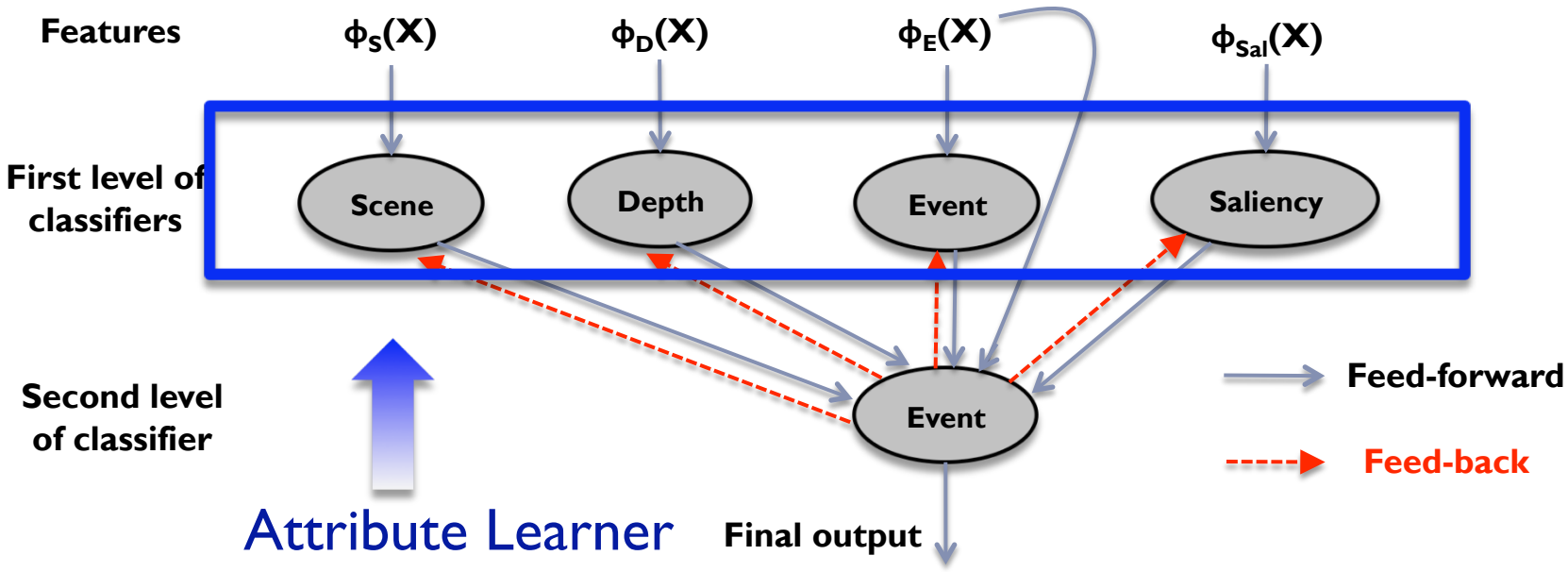
Motivation



- A model where the first layer is not trained to achieve the best independent performance, but achieve the best performance at the final output.

Model

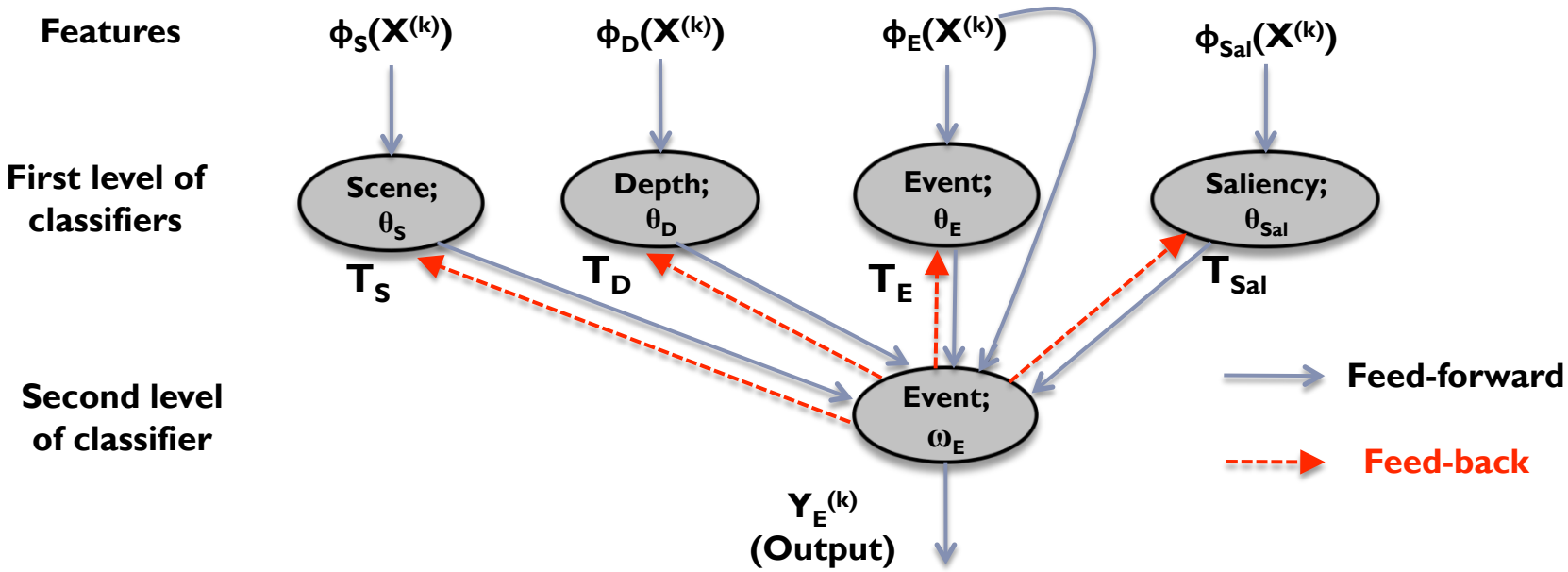
Model



- Proposed generic model enables composing "black-box" classifiers
- Feedback results in the first layer learning "attributes" rather than labels

Algorithm

Algorithm



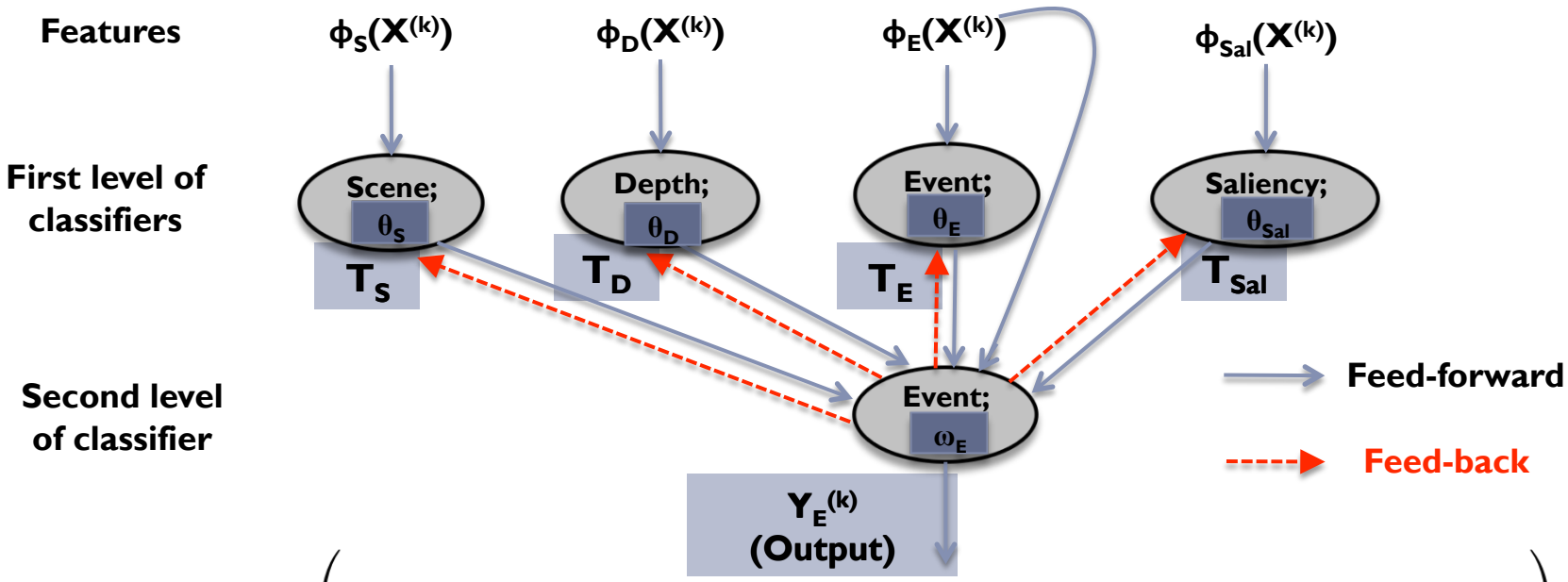
Optimization Goal

$$params^* = \underset{params}{\text{maximize}} \log \left(\prod_{all\ data} P(Output \mid Inputs\ ;\ params) \right)$$

$$\underset{\omega_E,\Theta}{\text{maximize}} \log \left(\prod_k P(Y_E^{(k)} \mid X^{(k)}\ ;\ \omega_E, \Theta) \right) \qquad \Theta = \{\theta_S, \theta_D, \theta_E, \theta_{Sal}\}$$

$$\underset{\omega_E,\Theta}{\text{maximize}} \sum_k \log \left(\sum_{T_S^{(k)},\ldots,T_{Sal}^{(k)}} \left(P(Y_E^{(k)} \mid X^{(k)}, T_S^{(k)}, \ldots, T_{Sal}^{(k)}\ ;\ \omega_E) \prod_{i \in S,\ldots,Sal} P(T_i^{(k)} \mid X^{(k)}\ ;\ \theta_i) \right) \right)$$

Algorithm

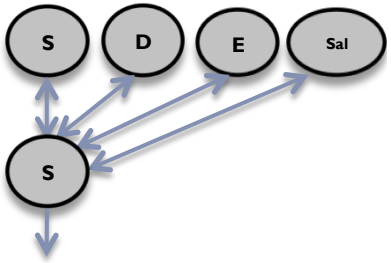


$$\underset{\omega_E,\Theta}{\text{maximize}} \sum_k \log \left(\sum_{T_S^{(k)},\ldots,T_{Sal}^{(k)}} \left(P(Y_E^{(k)} \mid X^{(k)}, T_S^{(k)}, \ldots, T_{Sal}^{(k)} \; ; \; \omega_E) \prod_{i \in S,\ldots,Sal} P(T_i^{(k)} \mid X^{(k)} \; ; \; \theta_i) \right) \right)$$

- **Our Solution:** Motivated from Expectation – Maximization (EM) algorithm
  - **Parameter Learning:** fix the required outputs and estimate parameters
  - **Latent Variable Estimation:** fix the model parameters and estimate latent variables (first level outputs)

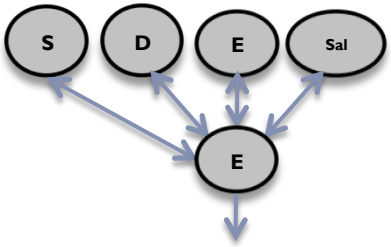
Results and Discussion

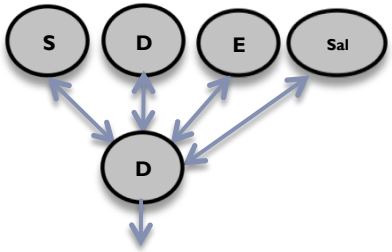
Experiments

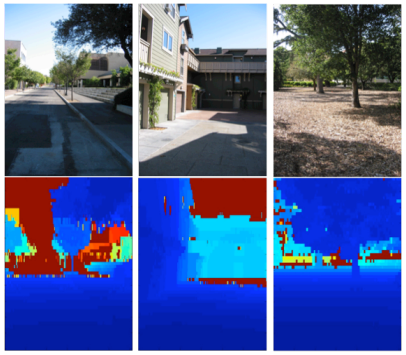


Scene Categorization
Oliva et al, IJCV'01

Event Categorization
Li et al, ICCV'07

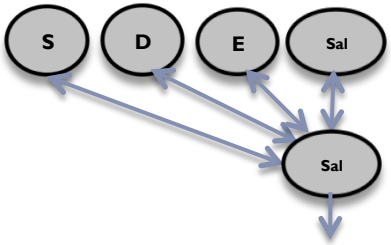






Depth Estimation - Make3D
Saxena et al, NIPS'05

Saliency Detection
Achanta et al, CVPR'09



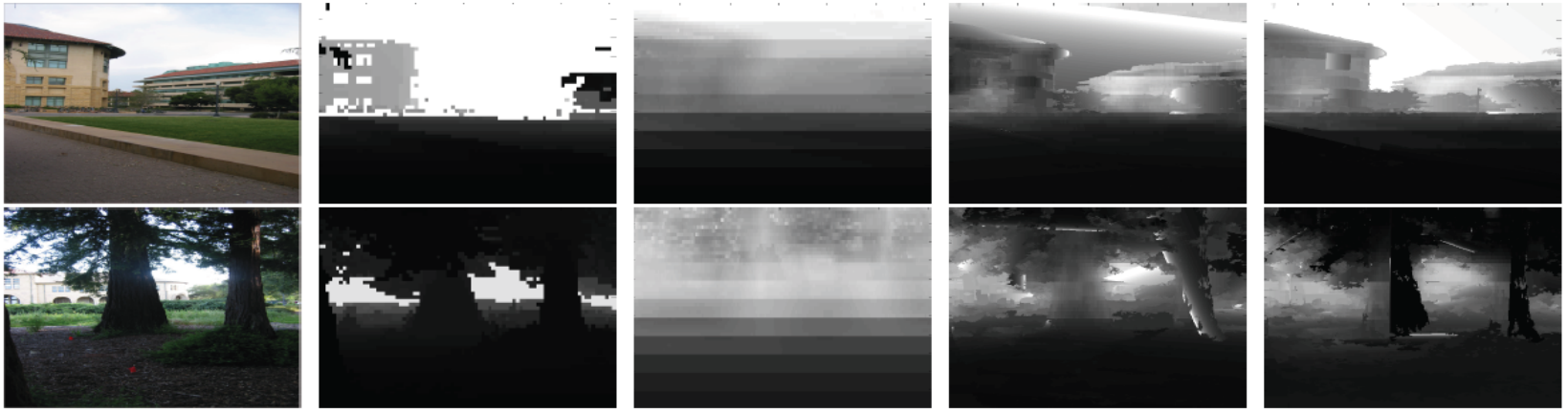
Results

Model	Event Categorization (% **Accuracy**)	Depth Estimation (RMS Error in m)	Scene Categorization (% **Accuracy**)	Saliency Detection (% **Accuracy**)
Images in testset	1579	400	2688	1000
Chance	12.5	24.6	12.5	50
State-of-art model	73.4 Li et.al. [14]	16.8 (MRF) Saxena et.al. [1]	83.7 Torralba et.al. [24]	82.5 (±0.2) Achanta et.al. [25]
Our base-model	72.0 (±0.8)	18.5 (±0.4)	83.8 (±0.2)	85.5 (±0.2)
All-features-direct	72.6 (±1.5)	16.4 (±0.4)	83.9 (±0.4)	86.2 (±0.2)
CCM (Heitz et.al.)	72.8 (±1.6)	16.2 (±0.4)	83.7 (±0.6)	86.5 (±0.2)
Sparse-CCM with feedback	**75.3 (±0.6)**	**15.2 (±0.1)**	**85.3 (±0.2)**	**87.3 (±0.1)**

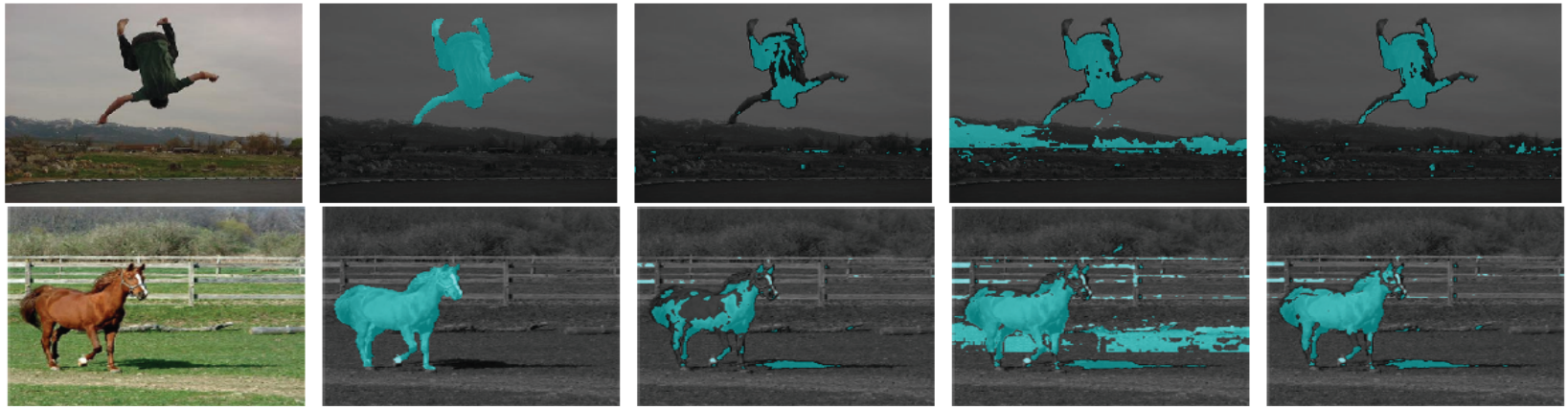
Improvement on every task with the same algorithm!

Results: Visual improvements

Depth Estimation



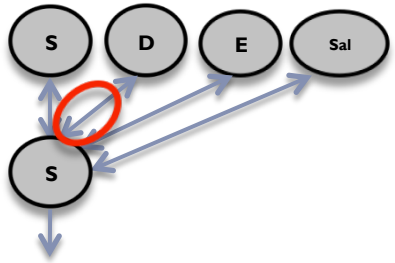
Saliency Detection

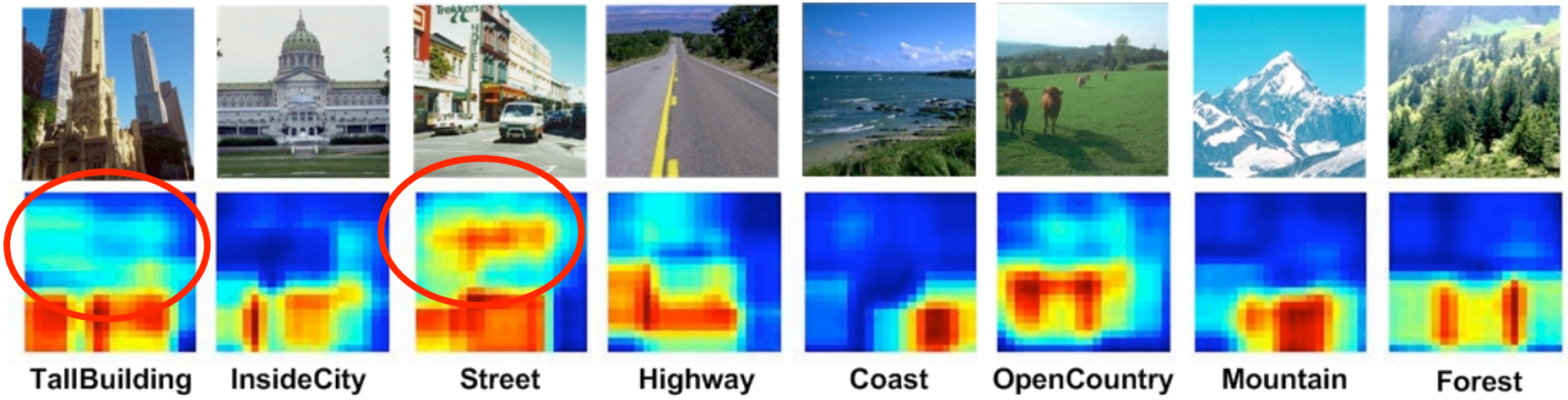


Original image | Ground truth | Base – model | CCM [Heitz et. al] | Our proposed

Discussion – Attributes of the scene

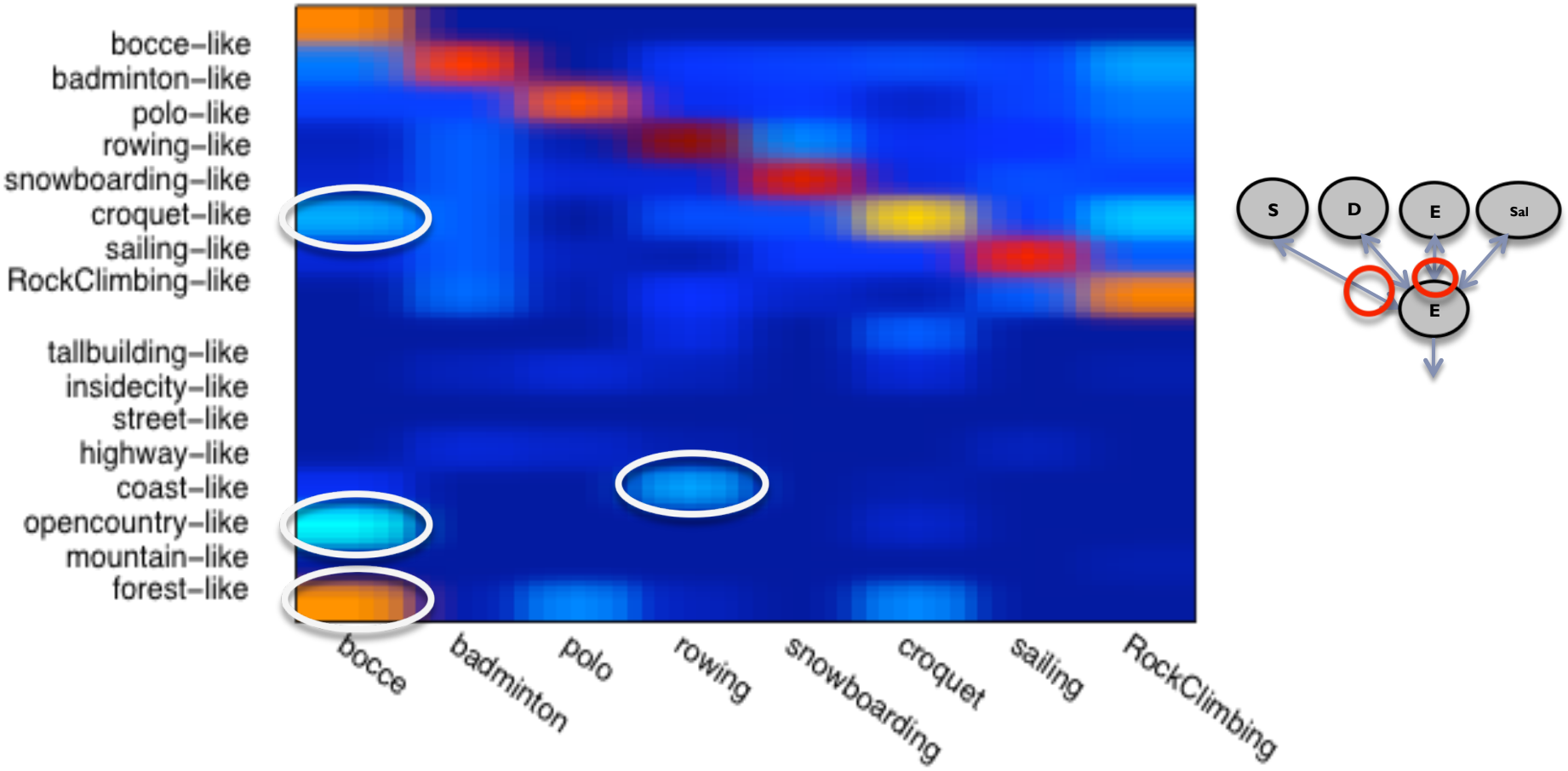
Maps of weights given to depth maps for scene categorization task





Discussion – Attributes of the scene

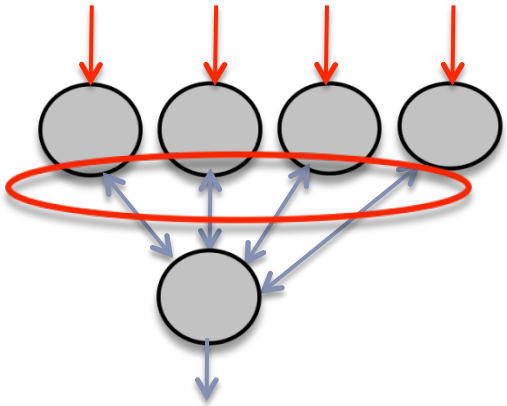
Weights given to event and scene attributes for event categorization



Conclusions

Conclusions

- Generic model to compose multiple vision tasks to aid holistic scene understanding
  - "Black-box"
- Feedback results in learning meaningful "attributes" instead of just the "labels"
- Handles heterogeneous datasets
- Improved performance for *each* of the tasks over state-of-art using the *same* learning algorithm
- Joint optimization of all the tasks?
  - Congcong Li, Adarsh Kowdle, Ashutosh Saxena, and Tsuhan Chen, *Feedback Enabled Cascaded Classification Models for Scene Understanding*, NIPS 2010



Thank you

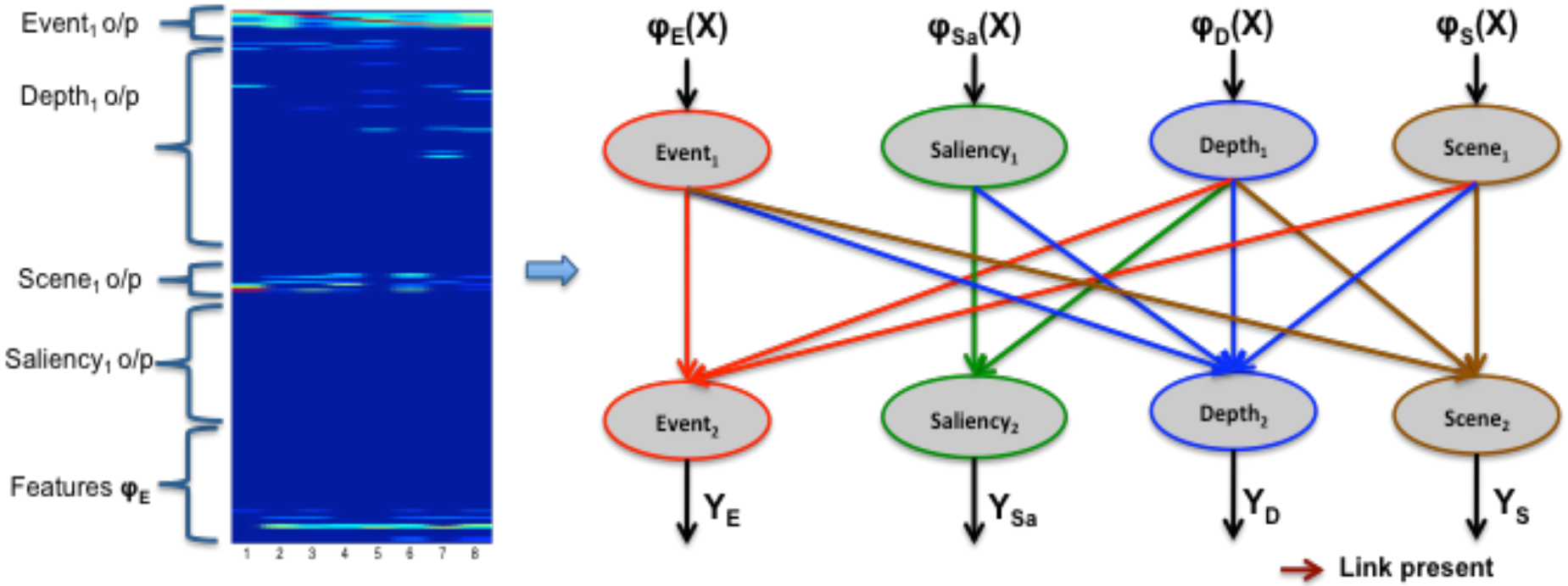
Questions?

Implementation

	Event Categorization	Depth Estimation	Scene Categorization	Saliency Detection
Image Feature Vector	51 – dim	104 – dim	512 – dim	3 – dim
1st layer Output	8 – dim class likelihood	Pixel level depth map	8 – dim class likelihood	Pixel level saliency score
Layer-1 Classifier	Multi-class Logistic	Linear Regression	RBF – kernel SVM	Linear Regression
Layer-2 Classifier	Multi-class Logistic	Linear Regression	Multi-class Logistic	Linear Regression

Discussion

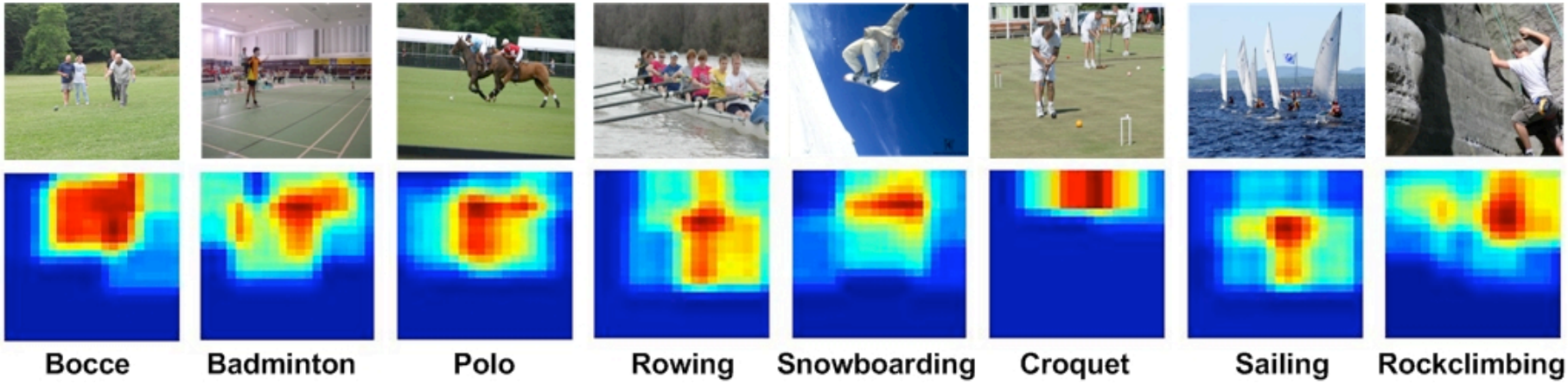
Sparse model learnt by our model



Weights for
event categorization task

Discussion – Attributes of the scene

Maps of weights given to depth maps for event categorization task



Results

Model	Event Categorization (% **Accuracy**)	Depth Estimation (RMS **Error** in m)	Scene Categorization (% **Accuracy**)	Saliency Detection (% **Accuracy**)
Images in testset	1579	400	2688	1000
Chance	12.5	24.6	12.5	50
State-of-art model	73.4 Li et.al. [14]	16.8 (MRF) Saxena et.al. [1]	83.7 Torralba et.al. [24]	82.5 (±0.2) Achanta et.al. [25]
Our base-model	72.0 (±0.8)	18.5 (±0.4)	83.8 (±0.2)	85.5 (±0.2)
All-features-direct	72.6 (±1.5)	16.4 (±0.4)	83.9 (±0.4)	86.2 (±0.2)
CCM (Heitz et.al.)	72.8 (±1.6)	16.2 (±0.4)	83.7 (±0.6)	86.5 (±0.2)
CCM with feedback	73.3 (±1.0)	15.3 (±0.2)	83.8 (±0.6)	87.3 (±0.2)
Sparse-CCM without feedback	73.6 (±1.4)	16.0 (±0.2)	83.9 (±0.2)	86.6 (±0.1)
Sparse-CCM with feedback	**75.3 (±0.6)**	**15.2 (±0.1)**	**85.3 (±0.2)**	**87.3 (±0.1)**

Improvement on every task with the same model!