Learning-Based PDE: A New Perspective for PDE Methods in Computer Vision

Risheng Liu

Dalian University of Technology

http://cggi.dlut.edu.cn/people/~rsliu/

rsliu@dlut.edu.cn

Outline

- **Motivation**
- Learning a specified PDE system
- Learning the general PDE system
- Discussions

PDE Methods in Computer Vision

- PDEs have been applied to
  - Denoising, Deblurring, Inpainting, Segmentation, Stereo, …
- Roughly have two kinds of methodologies for PDE design
  - Direct design: write down PDEs directly using some insights from physics

Now is Winter for PDE methods!!

Anisotropic Diffusion: $$\frac{\partial u}{\partial t} = \mathrm{div}(c(\|\nabla u\|)\nabla u),\ u|_{t=0} = f,\ \frac{\partial f}{\partial n}|_{\partial D} = 0$$

  - Variational design: energy functional $\longrightarrow$ Euler-Lagrange equation

Total Variation: $$\min_u \int_\Omega \|\nabla u\| + \lambda \|f - u\|^2$$

Not a hot topic in computer vision society RECENTLY!

Why PDEs fell out of favor?

- PDE is a powerful tool for data analysis
  - PDE is indeed a general regressor to approximate the nonlinear mapping between input and output vision data.
  - Differential system should be more efficient than linear equation for data analysis.
- Unfortunately…
  - It is challenging to design PDE in handcraft way (huge feasible solution set)
    - Deep insights for the problem and the data
    - High mathematical skills for building a PDE system
  - Fixed PDE formulation and/or boundary conditions is lack of flexibility

Beautiful mathematical formulation, BUT poor performance and limited application!!

理想很丰满、现实很骨感

New Perspective: **PDE** + **Learning**

Two possible ways to incorporate "learning" ideas into PDE designing:

- For a specific kind of PDE system (e.g., heat diffusion), we may adaptively learn its form and boundary for different data and tasks
  - **Risheng Liu**, Junjie Cao, Zhouchen Lin and Shiguang Shan. **Adaptive Partial Differential Equation Learning for Visual Saliency Detection**. CVPR'2014 (Oral)
- For using PDE to address general vision tasks, we may learn the PDE form from the "**dictionary of differential operators**" for different image processing and analysis tasks
  - **Risheng Liu**, Zhouchen Lin, Wei Zhang, Kewei Tang and Zhixun Su. **Toward Designing Intelligent PDEs for Computer Vision: An Optimal Control Approach**. Image and Vision Computing, 2013
  - **Risheng Liu**, Zhouchen Lin, Wei Zhang and Zhixun Su. **Learning PDEs for Image Restoration via Optimal Control**. ECCV'2010

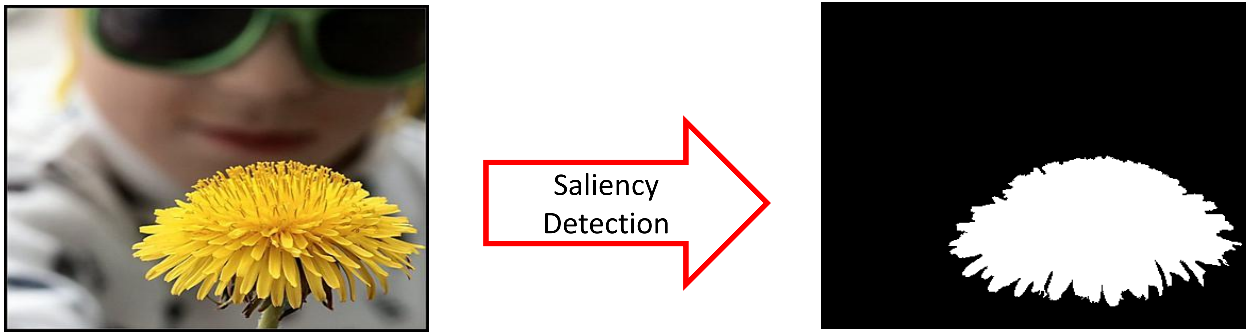
Outline

- Motivation
- **Learning a specified PDE system**
- Learning the general PDE system
- Discussions

Saliency Detection

Saliency detection is a hot topic in computer vision recently.

- Bottom-up methods: from image features
  - Feature + Clustering [Itti'98, Cheng'11, Lang'12, Shen'12, Wei'12, Xie'13]
  - Candidates + Refinement [Jiang'13, Yang'13]
- Top-down methods: from saliency model
  - CRF [Liu'11, Yang'12].



Input Image **Masked Salient Region**

Learning Heat Diffusion for Saliency Detection

- We assume that visual attention starts from saliency seeds and then propagates to a salient region.
- General heat diffusion system for image processing

Governing Equation

Boundary Condition

Initial Condition

$$
\begin{cases}
\dfrac{\partial f(\mathbf{p},t)}{\partial t} = F(f, \nabla f), & (\mathbf{p},t) \in \quad \times (0,T], \\
f(\mathbf{p},t) = g(t), & \mathbf{p} \in \partial \quad \times (0,T], \\
f(\mathbf{p},0) = I(\mathbf{p}). &
\end{cases}
$$

- But not for saliency detection!
- Difficulties of PDE methods
  - Hard to describe high-level information
  - Using fixed boundary conditions

Learn Governing Equation

Learn Boundary Conditions

$$\frac{\partial f(\mathbf{p},t)}{\partial t} = F(f, \nabla f), \quad s.t.\ \text{boundary conditions.}$$

Propagate visual attention from saliency seeds to other image elements

Local Diffusivity

We need a diffusion equation

We need Dirichlet boundary conditions

$$F(f, \nabla f) = \mathrm{div}(\boxed{\mathbf{K}_\mathbf{p}} \nabla f(\mathbf{p})) + \lambda(f(\mathbf{p}) - \boxed{g(\mathbf{p})}). \qquad f(\mathbf{g}) = 0, f(\mathbf{p}) = s_\mathbf{p}, \mathbf{p} \in \mathcal{S}.$$

Guidance Map

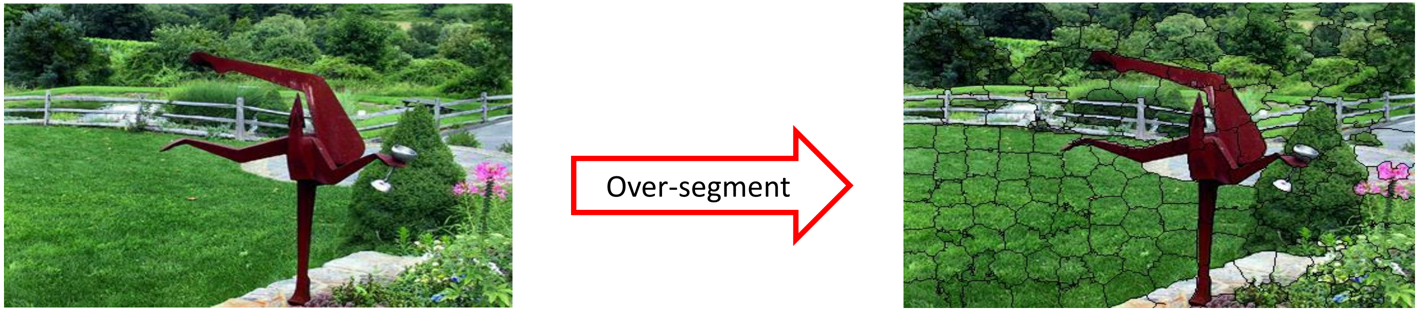
Steady state $\Rightarrow$ $F(f, \nabla f) = 0, \quad s.t. \quad f(\mathbf{g}) = 0, f(\mathbf{p}) = s_\mathbf{p}, \mathbf{p} \in \mathcal{S}.$

How to determine $\mathbf{K}_\mathbf{p}$, $g(\mathbf{p})$, and $s_\mathbf{p}$?

How to determine $\mathcal{S}$?

Learning PDEs – Graph Representation

- Undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$.
- Oversegment to superpixels: $\mathcal{V} = \{\mathbf{p}_1, \cdots, \mathbf{p}_{|\mathcal{V}|}\}$

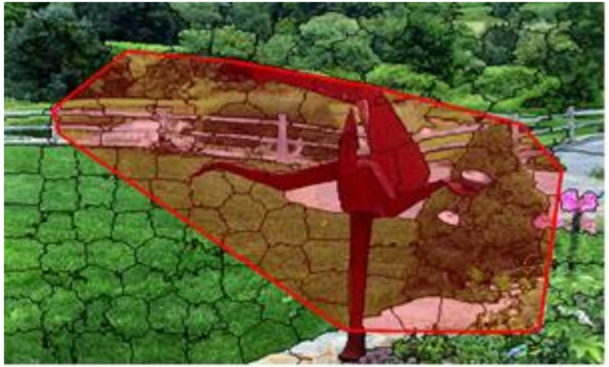


Input Image

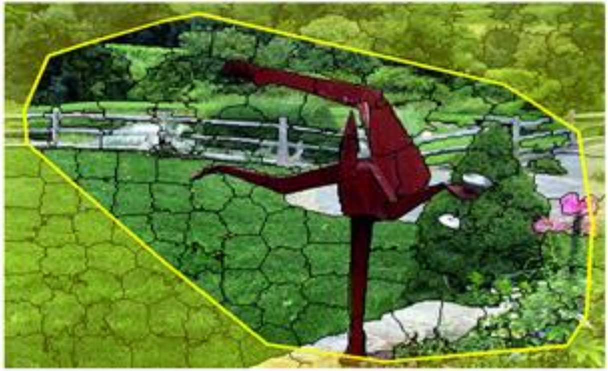
Superpixel Segmentation

Learning PDEs – Candidate Foreground and Background

- Use Harris operator to detect corners and contour points and estimate a convex hull C.
  - Candidate foreground $\mathcal{F}_c$ = nodes inside C.
- Add adjacent nodes to C to form an expanded hull C'.
  - Pure background $\mathcal{B}_c$ = nodes outside C'.



Candidate Foreground



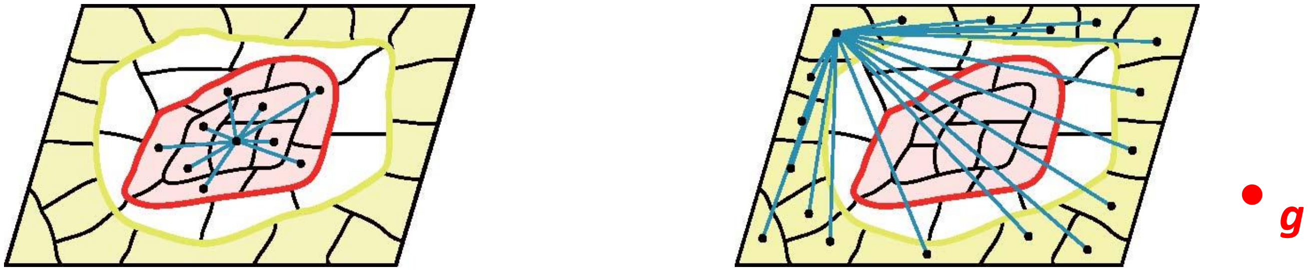
Candidate Background

Learning PDEs – Graph Construction

- Undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$.
  - Nodes in $\mathcal{F}_c$ within distance 2 are connected to each other (Fig. (a)).
  - All nodes in $\mathcal{B}_c$ are connected to each other: for smoothness of background (Fig. (b)).

$\boxed{\text{Defines neighborhood } \mathcal{N}_{\mathbf{p}}.}$

  - All nodes are connected an environment point $\mathbf{g}$.



(a) (b)

The red and yellow regions in (a) and (b) represent $\mathcal{F}_c$ and $\mathcal{B}_c$, respectively. Lines in (b) indicate that all nodes in $\mathcal{B}_c$ are connected to each other.

Learning PDEs – Learning $\mathbf{K_p}$

• Feature vectors: $\{h(\mathbf{p}), \mathbf{p} \in \mathcal{V}\}$ – mean of each superpixel in CIELAB color space.

$$\mathbf{K_p} = \mathrm{diag}(k(\mathbf{p}, \mathbf{q}_1), \cdots, k(\mathbf{p}, \mathbf{q}_{|\mathcal{N}_\mathbf{p}|-1}), \varepsilon_\mathbf{g}),$$

where $\mathcal{N}_\mathbf{p} = \{\mathbf{q}_1, \cdots, \mathbf{q}_{|\mathcal{N}_\mathbf{p}|-1}, \mathbf{g}\}$ is the neighborhood of $\mathbf{p}$,

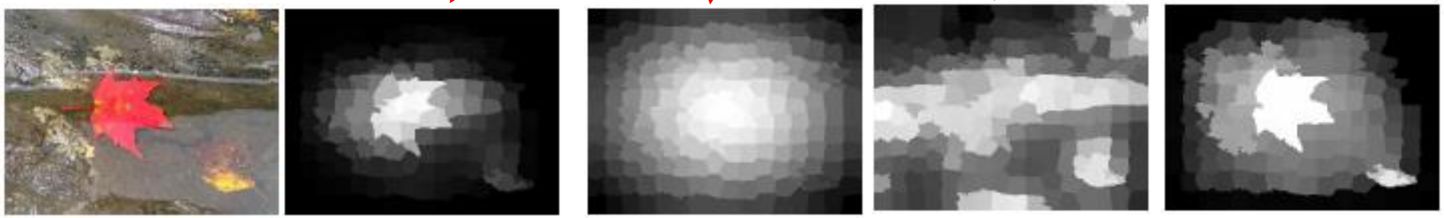
$$k(\mathbf{p}, \mathbf{q}) = \exp(-\beta \|h(\mathbf{p}) - h(\mathbf{q})\|^2),$$

and $0 < \varepsilon_\mathbf{g} \ll 1$ is the dissipation conductance at $\mathbf{p}$.

Learning PDEs – Learning $g(\mathbf{p})$ and $s_{\mathbf{p}}$

$$\boxed{g(\mathbf{p}) = f_l(\mathbf{p}) \times f_c(\mathbf{p}) \times f_f(\mathbf{p}),}$$

and then normalized.



- $f_c(\mathbf{p})$ and $f_l(\mathbf{p})$ are computed according to [Shen'12] and [Judd'09].
- $f_f(\mathbf{p}) = 1 - f_b(\mathbf{p})$: probability of belonging to foreground.

$$\mathrm{div}(\mathbf{K}_{\mathbf{p}} \nabla f_b(\mathbf{p})) = 0, \quad s.t. \quad f_b(\mathbf{g}) = 0, f_b(\mathbf{p}) = 1, \mathbf{p} \in \mathcal{B}_c.$$

$$\boxed{s_{\mathbf{p}} = g(\mathbf{p}), \text{ for } \mathbf{p} \in \mathcal{S}.}$$

X. Shen and Y. Wu. A unified approach to salient object detection via low rank matrix recovery. In CVPR, 2012.
T. Judd, K. Ehinger, F. Durand, and A. Torralba. Learning to predict where humans look. In ICCV, 2009.

Learning PDEs – Learning *S*

• Optimize saliency seeds by searching for the most representative foreground nodes in $\mathcal{F}_c$ as $\mathcal{S}$.

• Maximize the sum of scores f w.r.t. all image elements in $\mathcal{V}$.

$$\max_{\mathcal{S}\subset\mathcal{F}_c,|\mathcal{S}|\leq n} L(\mathcal{S}) \equiv \sum_{\mathbf{p}\in\mathcal{V}} f(\mathbf{p}),$$

$$s.t. \begin{cases} \mathrm{div}(\mathbf{K_p}\nabla f(\mathbf{p})) + \lambda(f(\mathbf{p}) - g(\mathbf{p})) = 0, \\ f(\mathbf{g}) = 0, f(\mathbf{p}) = s_{\mathbf{p}}, \mathbf{p} \in \mathcal{S}. \end{cases}$$

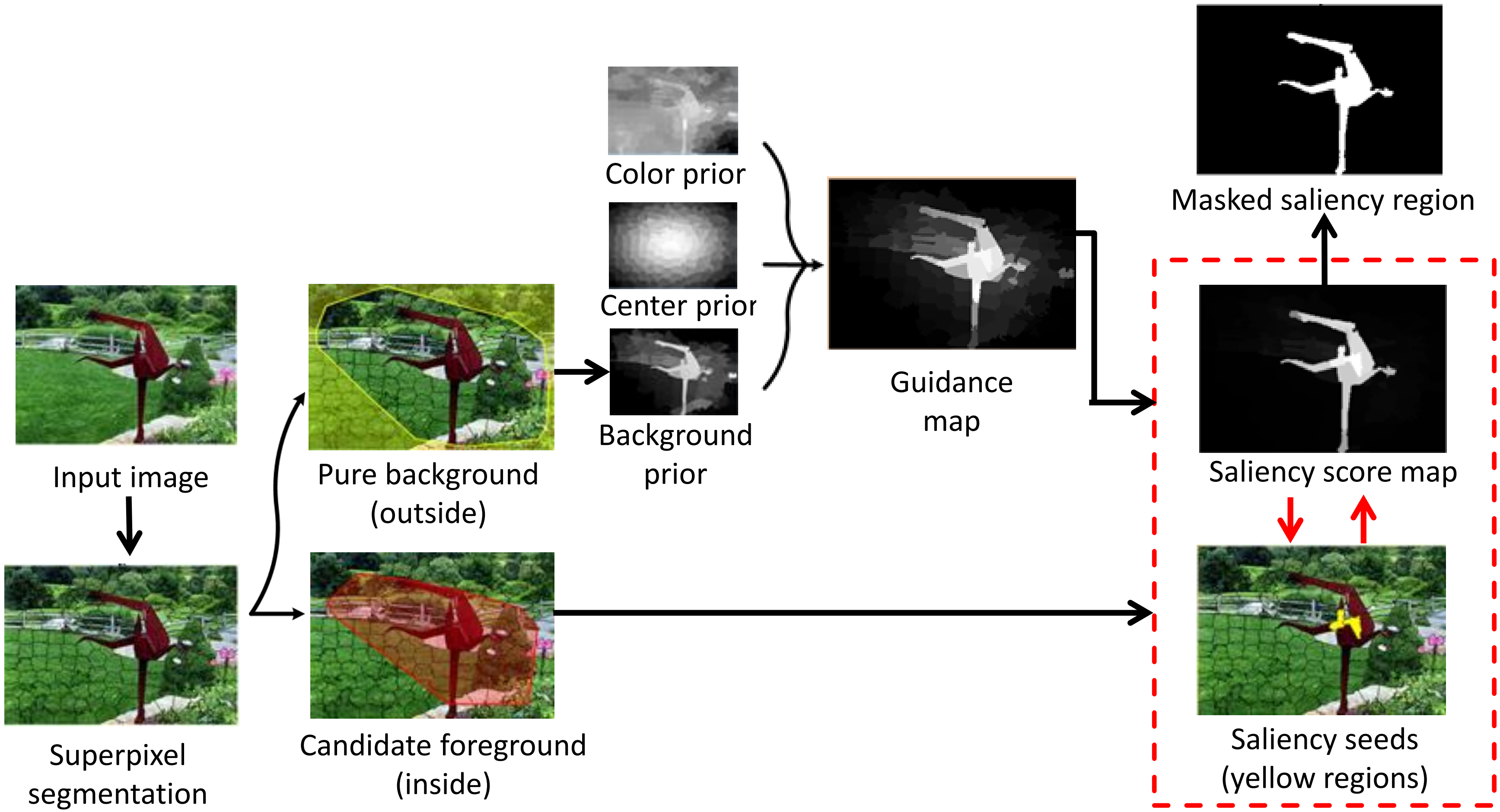
PDE Constrained Discrete Optimal Control

Theorem: $L(\mathcal{S})$ is submodular and monotone w.r.t. $\mathcal{S}$.

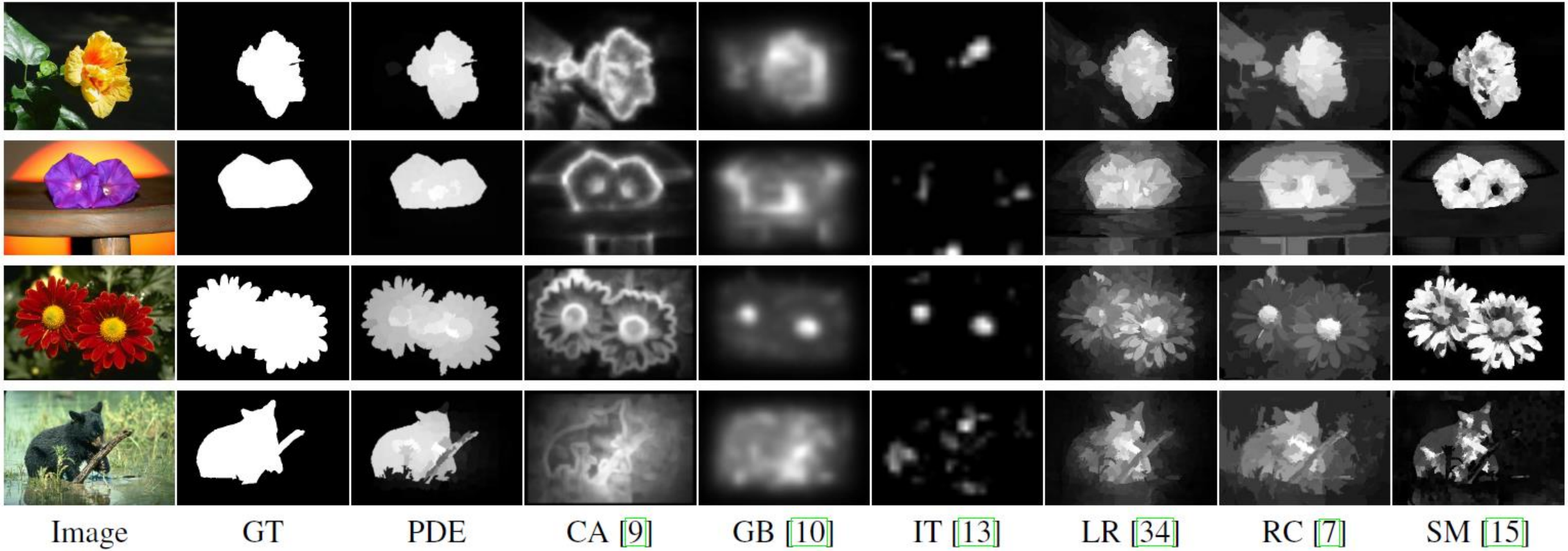
⇩

$\boxed{\text{Greedy algorithm works well.}}$

The Pipeline of PDE Based Saliency Detection

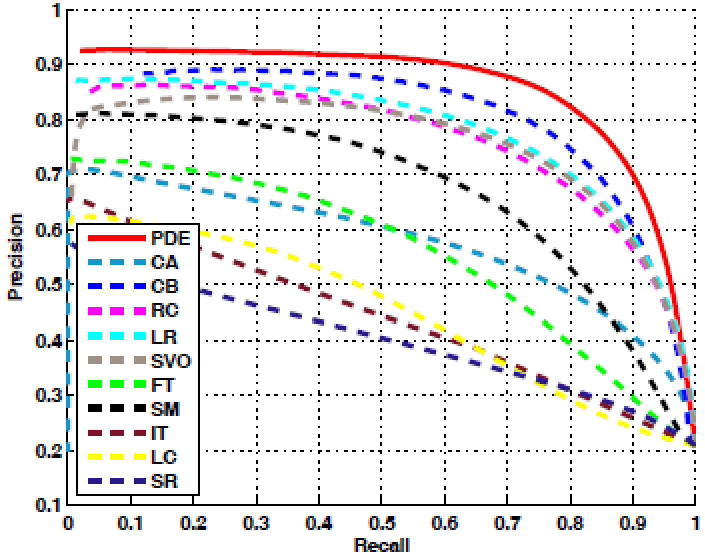


Experiments – Qualitative Comparison

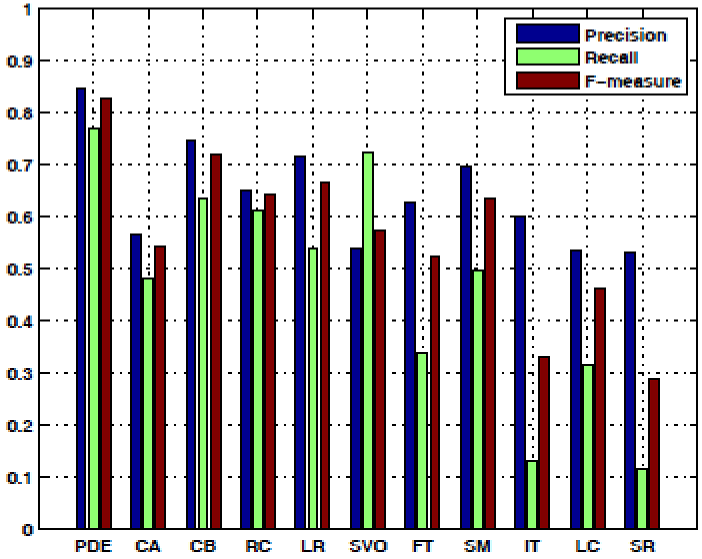


The top three rows are examples in MSRA and the bottom is in Berkeley.

Experiments – Quantitative Comparisons



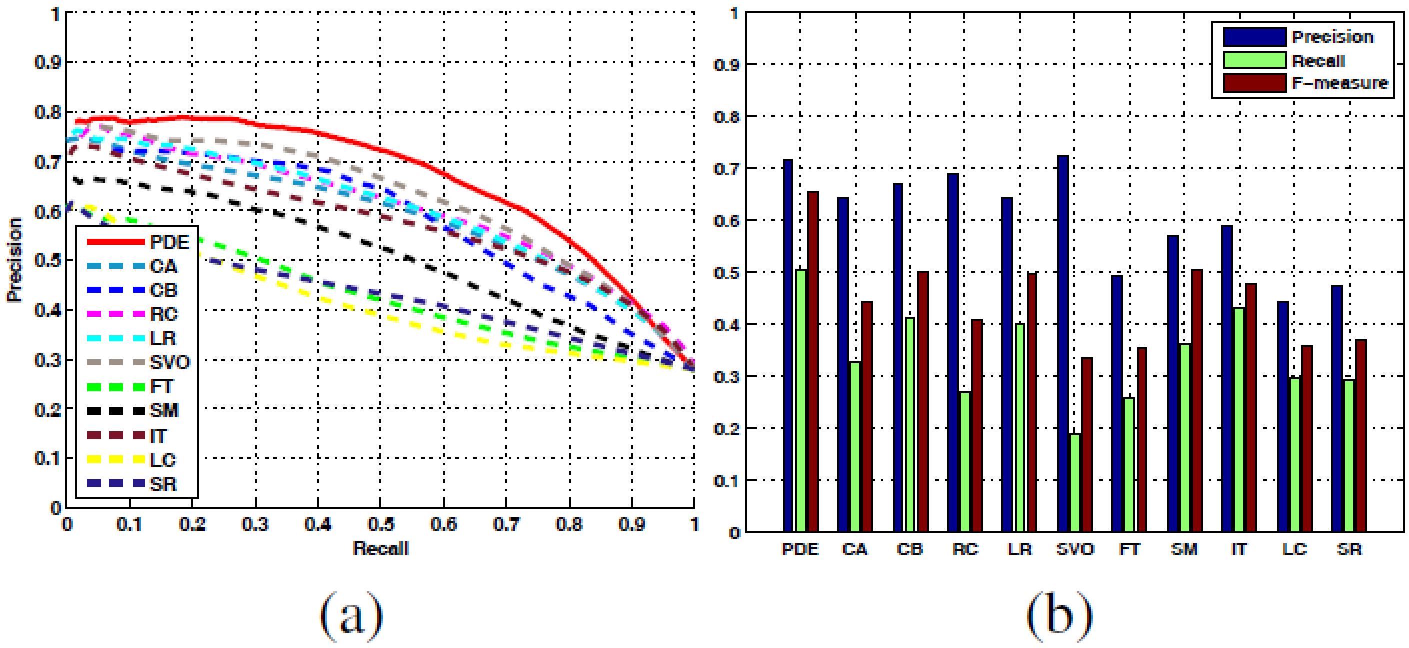
(a)



(b)

Results on the **MSRA-5000** image set. (a) Precision-recall curves. (b) Average precision, recall, and F-measure values.

Experiments – Quantitative Comparisons



Results on the **Berkeley** image set. (a) Precision-recall curves. (b) Average precision, recall, and F-measure values.

Summary

- Propose an adaptive PDE to model the saliency diffusion.
  - The first PDE based method for saliency detection.
  - Learn both the governing equation and the boundary condition of the PDE.
- Optimize saliency seeds by submodular maximization.
  - Saliency seeds make the boundary condition.
- Learning a specific PDE system for more vision and graphics tasks (Ongoing)
  - Robust object tracking
  - Mesh segmentation and labeling

Outline

- Motivation
- Learning a specified PDE system
- **Learning the general PDE system**
- Discussions

Can we just consider PDEs as a **BLACK BOX** for computer vision?

- A user only have to prepare input/output training data!
- The same framework for various problems!
- Can we have such a convenient way?

Possible!!!

Learning a general PDE system for computer vision

- Design a general "**dictionary**" of differential operators
- Learn combination "**codes**" for specific task and/or data

General PDE Formulation

Most existing evolutionary PDEs for a image u can be brought to as follows:

$$\frac{\partial u}{\partial t} - F(u, \nabla u, \mathbf{H}_u) = 0,$$

where F is the function of u, ∇u and $\mathbf{H}_u$.

- The space of PDEs is infinitely dimensional
- The designing of PDEs is heavily rely on intuition
- It is hard to quantify and describe the intuition
- The designed PDEs can only reflect very limited aspects of a vision task

Dictionary for PDEs in Computer Vision
(Shift/Rotation Invariant Fundamental Differential Invariants)

j	$\mathrm{inv}_j(u,v)$
0,1,2	$1,\ v,\ u$
3,4	$\Vert\nabla v\Vert^2 = v_x^2 + v_y^2,\ \Vert\nabla u\Vert^2 = u_x^2 + u_y^2$
5	$(\nabla v)^T\nabla u = v_x u_x + v_y u_y$
6,7	$\mathrm{tr}(\mathbf{H}_v) = v_{xx} + v_{yy},\ \mathrm{tr}(\mathbf{H}_u) = u_{xx} + u_{yy}$
8	$(\nabla v)^T\mathbf{H}_v\nabla v = v_x^2 v_{xx} + 2v_x v_y v_{xy} + v_y^2 v_{yy}$
9	$(\nabla v)^T\mathbf{H}_u\nabla v = v_x^2 u_{xx} + 2v_x v_y u_{xy} + v_y^2 u_{yy}$
10	$(\nabla v)^T\mathbf{H}_v\nabla u = v_x u_x v_{xx} + (v_x u_y + v_y u_x)v_{xy} + v_y u_y v_{yy}$
11	$(\nabla v)^T\mathbf{H}_u\nabla u = v_x u_x u_{xx} + (v_x u_y + v_y u_x)u_{xy} + v_y u_y u_{yy}$
12	$(\nabla u)^T\mathbf{H}_v\nabla u = u_x^2 v_{xx} + 2u_x u_y v_{xy} + u_y^2 v_{yy}$
13	$(\nabla u)^T\mathbf{H}_u\nabla u = u_x^2 u_{xx} + 2u_x u_y u_{xy} + u_y^2 u_{yy}$
14	$\mathrm{tr}(\mathbf{H}_v^2) = v_{xx}^2 + 2v_{xy}^2 + v_{yy}^2$
15	$\mathrm{tr}(\mathbf{H}_v\mathbf{H}_u) = v_{xx}u_{xx} + 2v_{xy}u_{xy} + v_{yy}u_{yy}$
16	$\mathrm{tr}(\mathbf{H}_u^2) = u_{xx}^2 + 2u_{xy}^2 + u_{yy}^2$

Linear Combination of Differential Invariants

$$F_u(u, v, \mathbf{a}) = \sum_{j=0}^{16} a_j(t)\mathrm{inv}_j(u, v),$$
$$F_v(v, u, \mathbf{b}) = \sum_{j=0}^{16} b_j(t)\mathrm{inv}_j(v, u).$$

where u is the target image and v is a indicator function for collecting large scale information.

Learning Coefficients by Optimal Control

$$\min J(\{u_m\}_{m=1}^M, \{a_j\}_{j=0}^{16}, \{b_j\}_{j=0}^{16})$$
$$= \sum_{m=1}^{M} \int_\Omega [u_m(x,y,T) - O_m]^2 \mathrm{d}\Omega + \sum_{j=0}^{16} \lambda_j \int_0^T a_j^2(t)\mathrm{d}t + \sum_{j=0}^{16} \mu_j \int_0^T b_j^2(t)\mathrm{d}t,$$

$$s.t. \quad \begin{cases} \frac{\partial u_m}{\partial t} - F_u(u_m, v_m, \{a_j\}_{j=0}^{16}) = 0, & (x,y,t) \in Q, \\ u_m(x,y,t) = 0, & (x,y,t) \in \Gamma, \\ u_m|_{t=0} = I_m, & (x,y) \in \Omega, \\ \frac{\partial v_m}{\partial t} - F_v(v_m, u_m, \{b_j\}_{j=0}^{16}) = 0, & (x,y,t) \in Q, \\ v_m(x,y,t) = 0, & (x,y,t) \in \Gamma, \\ v_m|_{t=0} = I_m, & (x,y) \in \Omega. \end{cases}$$

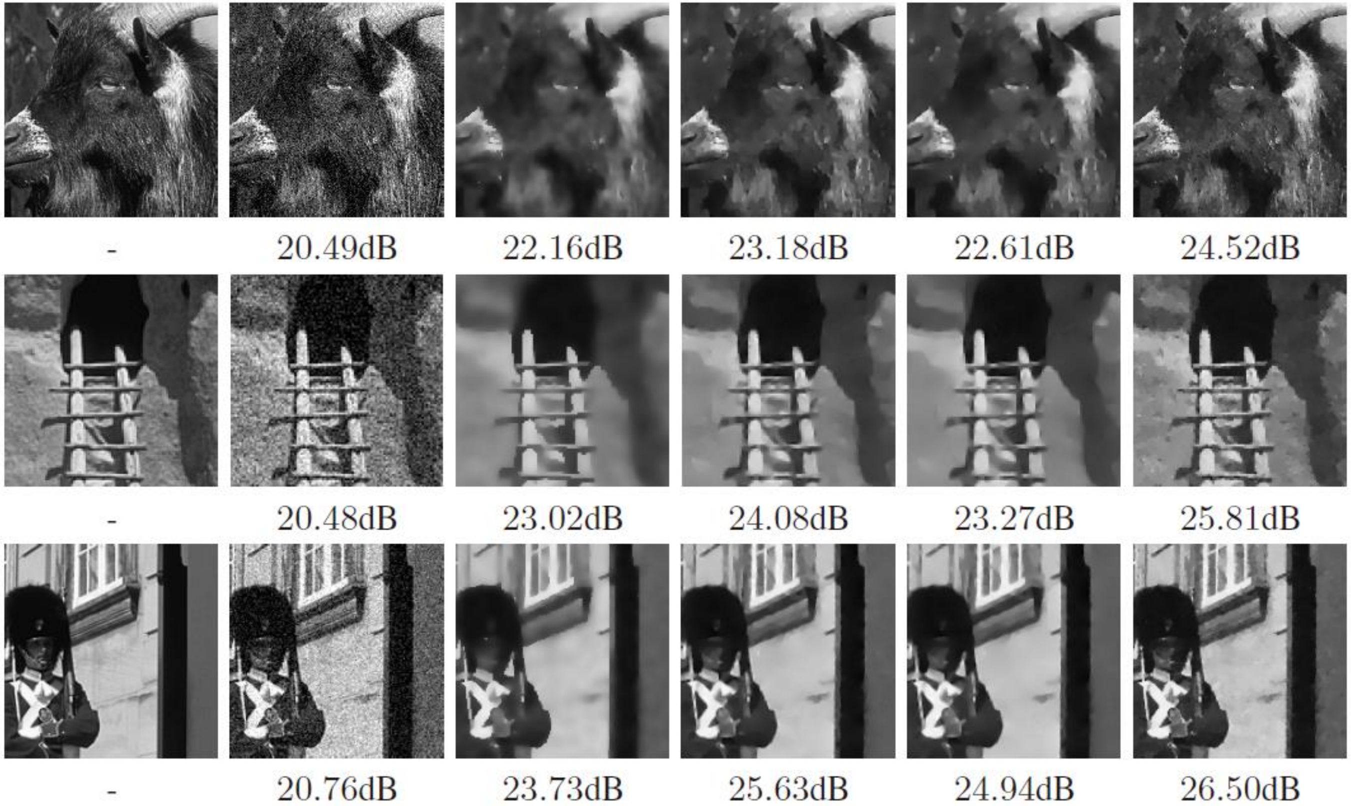
where (I_m, O_m) are training samples, where I_m is the input image and O_m is the expected output image, $m = 1, \cdots, M$.

Experiments

- The same form of PDE for different problems!
  - Only differ in coefficients
- Some of them have never been tried by PDEs!

Risheng Liu, Zhouchen Lin, Wei Zhang, Kewei Tang and Zhixun Su. Toward Designing Intelligent PDEs for Computer Vision: An Optimal Control Approach. ***Image and Vision Computing***, 2013
Risheng Liu, Zhouchen Lin, Wei Zhang and Zhixun Su. Learning PDEs for Image Restoration via Optimal Control. ***ECCV'2010***

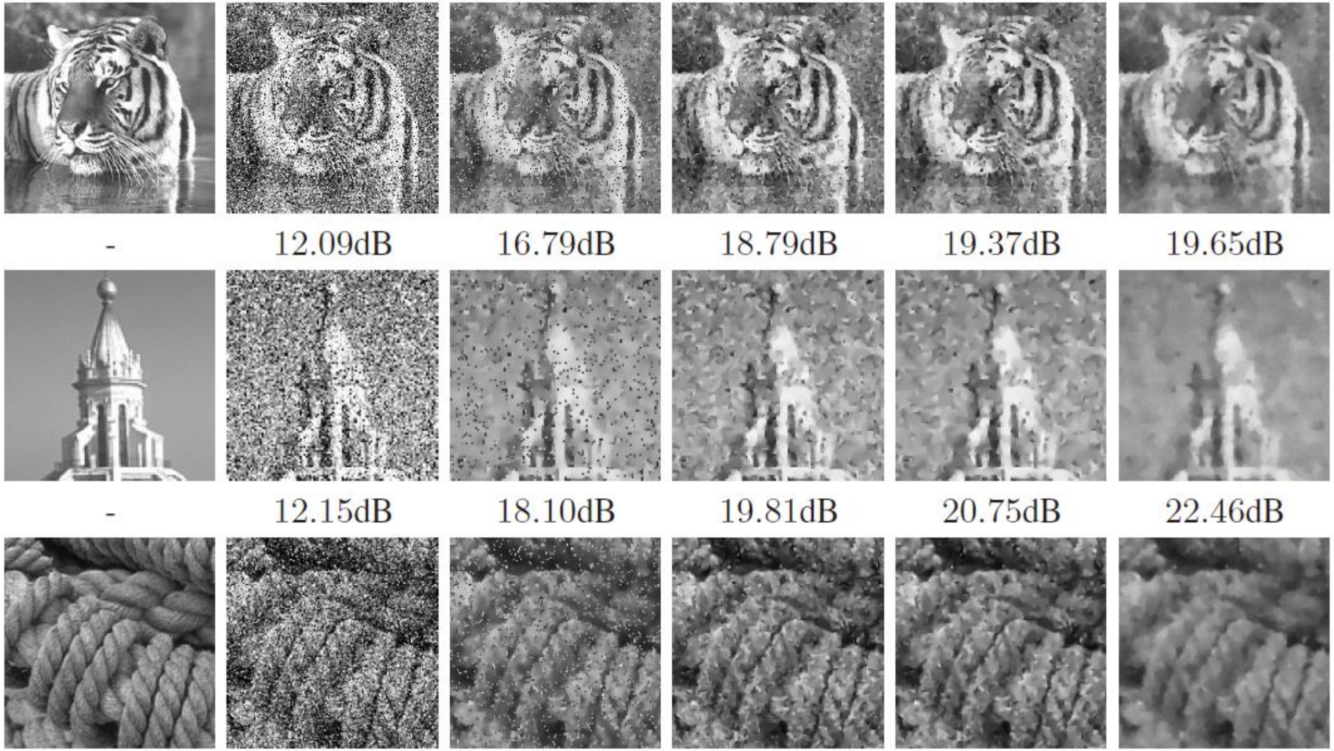
Denoising (Gaussian)



(a) original (b) noisy (c) P-M (d) ROF (e) TV-L^1 (f) L-PDE

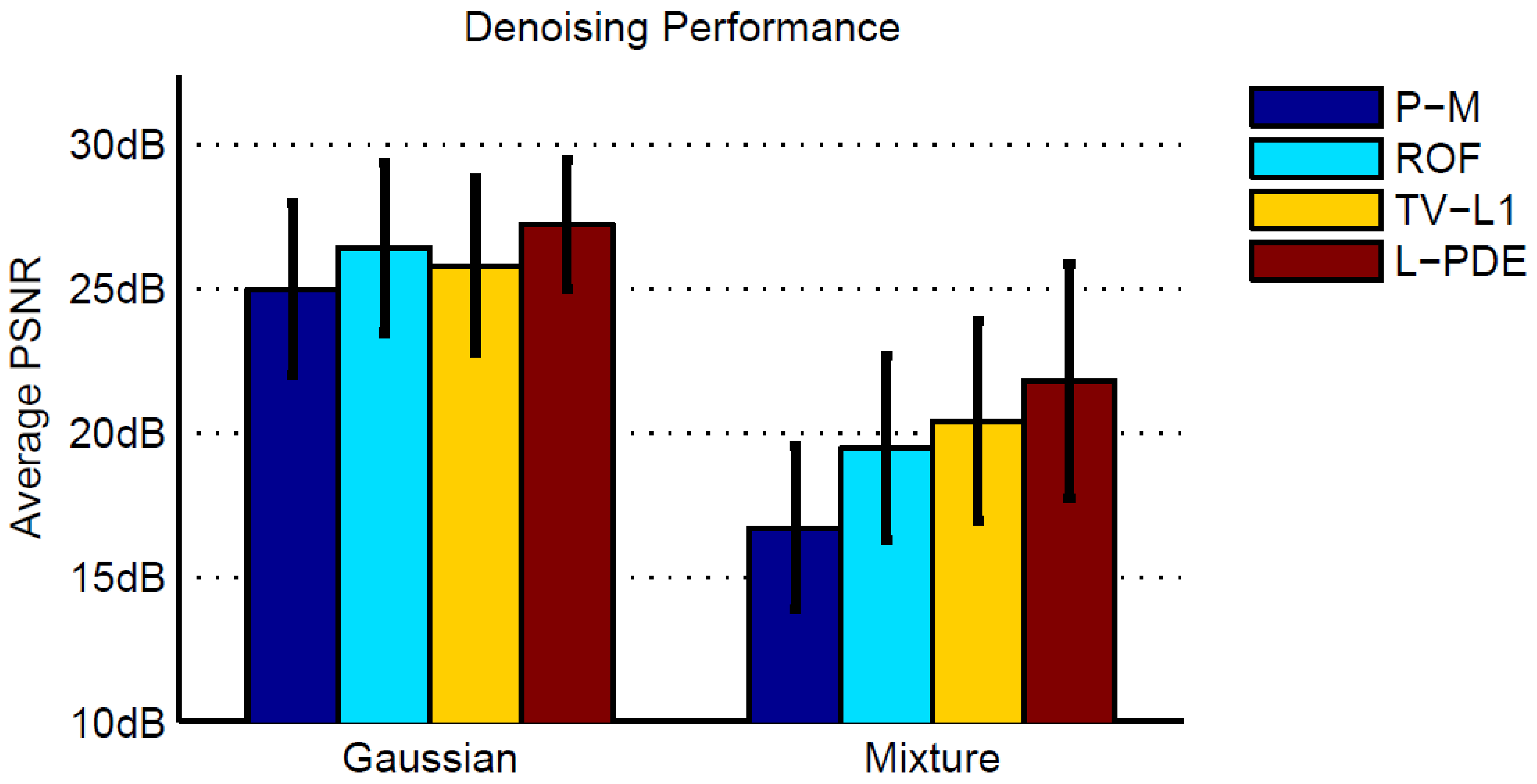
Risheng Liu, Zhouchen Lin, Wei Zhang and Zhixun Su. Learning PDEs for Image Restoration via Optimal Control. ***ECCV'2010***

Denoising (Mixture of Gaussian, Poisson and Salt & Pepper)



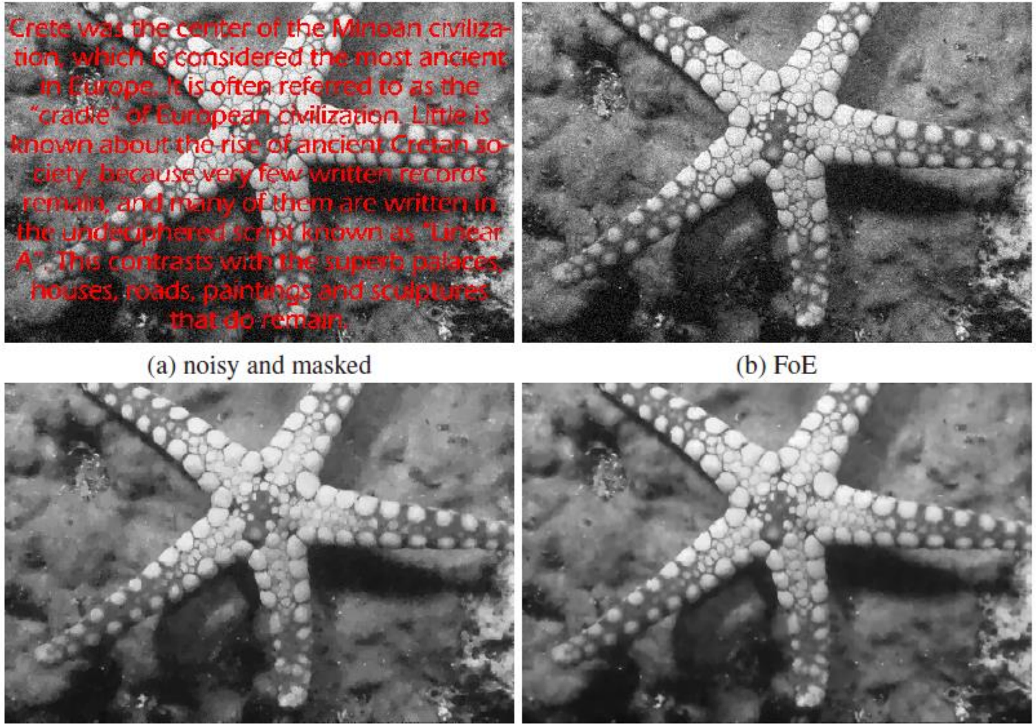
(a) original	(b) noisy	(c) P-M	(d) ROF	(e) TV-L^1	(f) L-PDE
-	12.09dB	16.79dB	18.79dB	19.37dB	19.65dB
-	12.15dB	18.10dB	19.81dB	20.75dB	22.46dB
-	12.29dB	18.35dB	19.53dB	20.26dB	21.26dB

Risheng Liu, Zhouchen Lin, Wei Zhang and Zhixun Su. Learning PDEs for Image Restoration via Optimal Control. ***ECCV'2010***



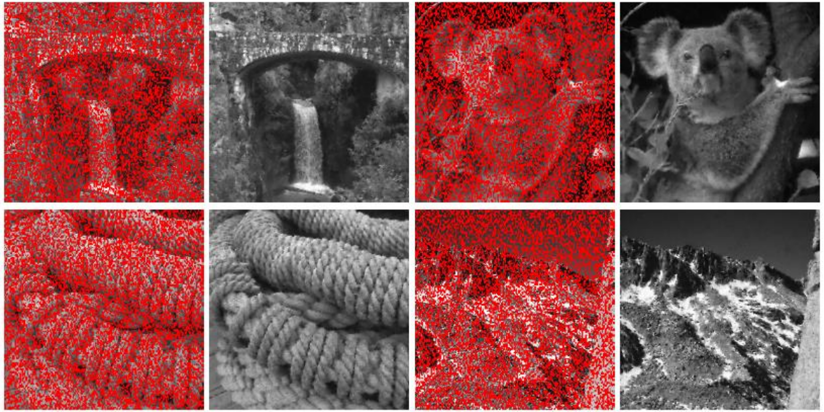
Risheng Liu, Zhouchen Lin, Wei Zhang and Zhixun Su. Learning PDEs for Image Restoration via Optimal Control. ***ECCV'2010***

Inpainting



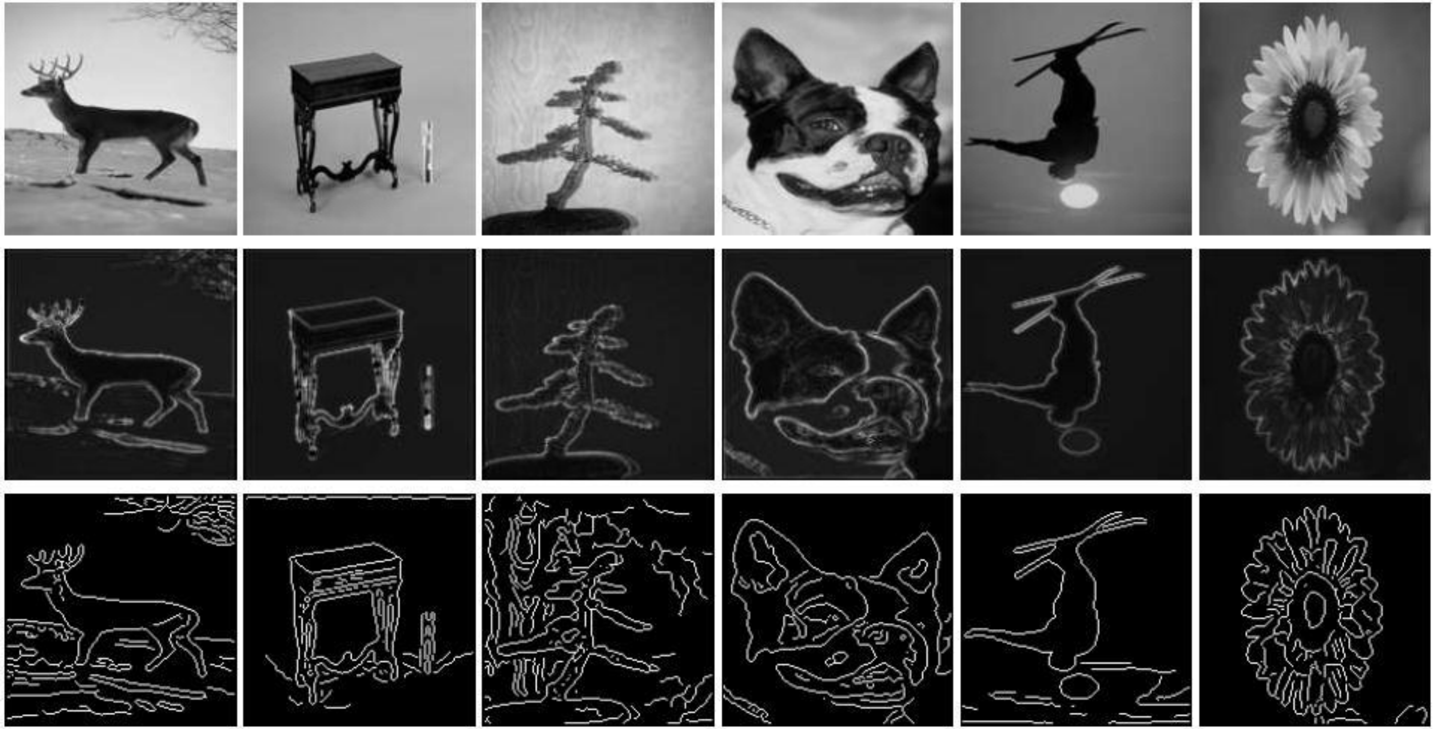
(a) noisy and masked (b) FoE

(c) TV (d) L-PDE



Risheng Liu, Zhouchen Lin, Wei Zhang and Zhixun Su. Learning PDEs for Image Restoration via Optimal Control. ***ECCV'2010***

Perceptual Edge Detection



Risheng Liu, Zhouchen Lin, Wei Zhang, Kewei Tang and Zhixun Su. Toward Designing Intelligent PDEs for Computer Vision: An Optimal Control Approach. ***Image and Vision Computing***, 2013

Object Detection (Training)

Risheng Liu, Zhouchen Lin, Wei Zhang, Kewei Tang and Zhixun Su. Toward Designing Intelligent PDEs for Computer Vision: An Optimal Control Approach. ***Image and Vision Computing***, 2013

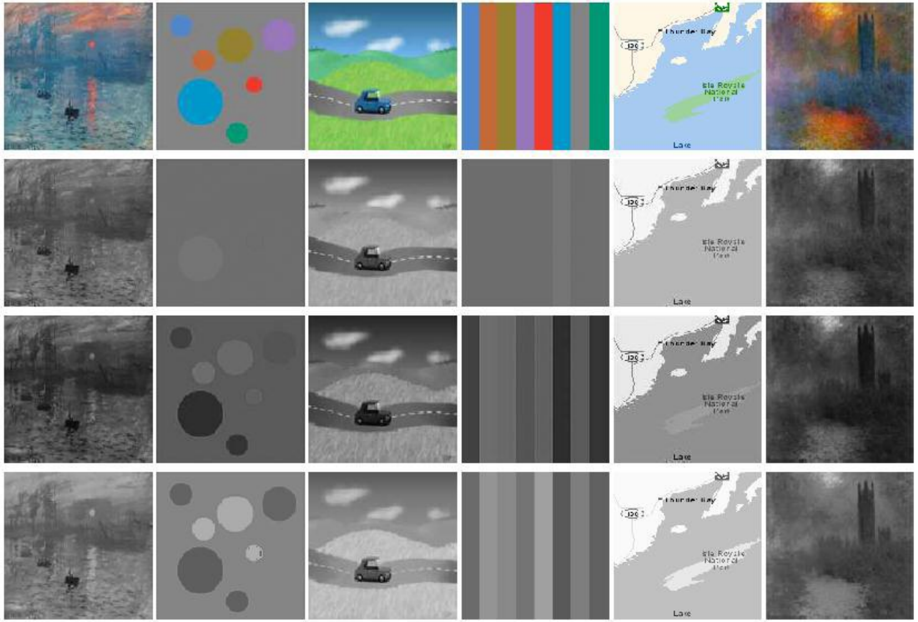
Object Detection (Testing, Positive)

Risheng Liu, Zhouchen Lin, Wei Zhang, Kewei Tang and Zhixun Su. Toward Designing Intelligent PDEs for Computer Vision: An Optimal Control Approach. ***Image and Vision Computing***, 2013

Object Detection (Testing, Negative)

Risheng Liu, Zhouchen Lin, Wei Zhang, Kewei Tang and Zhixun Su. Toward Designing Intelligent PDEs for Computer Vision: An Optimal Control Approach. ***Image and Vision Computing***, 2013

Color Image Processing



Color2gray

Demosaicking

Risheng Liu, Zhouchen Lin, Wei Zhang, Kewei Tang and Zhixun Su. Toward Designing Intelligent PDEs for Computer Vision: An Optimal Control Approach. ***Image and Vision Computing***, 2013

Outline

- Motivation
- Learning a specified PDE system
- Learning the general PDE system
- **Discussions**

The spring of Neural Network is coming, but when is PDE?

- Feature learning beats handcraft features
- Can we do similar thing for PDE methods?
  - Building blocks of DNN are linear systems, while PDE considers more powerful differential systems
  - The evolution of PDE is very similar to the multiple layers NN learning process

Take Home Message

- PDE + Learning is a promising direction for computer vision
  - Potential connections to deep learning?!
- Learning a specific PDE for particular task
- Learning a general PDE for various tasks
- Build "deep" PDE system for challenging tasks?

References

- **Risheng Liu**, Junjie Cao, Zhouchen Lin and Shiguang Shan. Adaptive Partial Differential Equation Learning for Visual Saliency Detection. ***CVPR' 2014*** (Oral)
- **Risheng Liu**, Zhouchen Lin, Wei Zhang, Kewei Tang and Zhixun Su. Toward Designing Intelligent PDEs for Computer Vision: An Optimal Control Approach. ***Image and Vision Computing***, 2013
- **Risheng Liu**, Zhouchen Lin, Wei Zhang and Zhixun Su. Learning PDEs for Image Restoration via Optimal Control. ***ECCV'2010***

小广告

- 大连理工大学数字媒体技术团队招生（计算数学、计算机科学与技术和软件工程专业博士、硕士）
- 欢迎各位老师同学以任何形式与我们开展交流、合作
- 诚邀各位CV同仁到大工、到软件学院指导、访问

Thanks for attentions

http://cggi.dlut.edu.cn/people/~rsliu/
rsliu@dlut.edu.cn