Efficient Linear Programming for Dense CRFs

Thalaiyasingam Ajanthan Alban Desmaison
Rudy Bunel Mathieu Salzmann Philip H. S. Torr
M. Pawan Kumar

Oxford, December 2016

Dense CRF

- Fully connected CRF with Gaussian pairwise potentials.

$$E(\mathbf{x}) = \sum_{a=1}^{n} \phi_a(x_a) + \sum_{a=1}^{n} \sum_{\substack{b=1 \\ b \neq a}}^{n} \psi_{ab}(x_a, x_b) \ ,$$

$$\psi_{ab}(x_a, x_b) = \underbrace{\mu(x_a, x_b)}_{\text{Label compatibility}} \underbrace{\exp\left(\frac{-\|\mathbf{f}_a - \mathbf{f}_b\|^2}{2}\right)}_{\text{Pixel compatibility}} ,$$

where $x_a \in \mathcal{L}$ and $\mathbf{f}_a \in \mathbb{R}^{\mathrm{d}}$.

Why?

- Captures long-range interactions and provides fine grained segmentations [Krähenbühl-2011].

Dense CRF

- Fully connected CRF with Gaussian pairwise potentials.

$$E(\mathbf{x}) = \sum_{a=1}^{n} \phi_a(x_a) + \sum_{a=1}^{n} \sum_{\substack{b=1 \\ b \neq a}}^{n} \psi_{ab}(x_a, x_b) \ ,$$

$$\psi_{ab}(x_a, x_b) = \underbrace{\mu(x_a, x_b)}_{\text{Label compatibility}} \underbrace{\exp\left(\frac{-\|\mathbf{f}_a - \mathbf{f}_b\|^2}{2}\right)}_{\text{Pixel compatibility}} \ ,$$

where $x_a \in \mathcal{L}$ and $\mathbf{f}_a \in \mathbb{R}^{\mathrm{d}}$.

Why?

- Captures long-range interactions and provides fine grained segmentations [Krähenbühl-2011].

Dense CRF

$$E(\mathbf{x}) = \sum_{a=1}^{n} \phi_a(x_a) + \sum_{a=1}^{n} \sum_{\substack{b=1 \\ b \neq a}}^{n} \psi_{ab}(x_a, x_b) \ ,$$

$$\psi_{ab}(x_a, x_b) = \mu(x_a, x_b) \exp\left(\frac{-\|\mathbf{f}_a - \mathbf{f}_b\|^2}{2} \right) \ .$$

Difficulty

- Requires $\mathcal{O}(n^2)$ computations $\Rightarrow$ Infeasible.

Idea

- Approximate using the filtering method [Adams-2010] $\Rightarrow \mathcal{O}(n)$ computations.

Dense CRF

$$E(\mathbf{x}) = \sum_{a=1}^{n} \phi_a(x_a) + \sum_{a=1}^{n} \sum_{\substack{b=1 \\ b \neq a}}^{n} \psi_{ab}(x_a, x_b) \ ,$$

$$\psi_{ab}(x_a, x_b) = \mu(x_a, x_b) \exp\left(\frac{-\|\mathbf{f}_a - \mathbf{f}_b\|^2}{2} \right) \ .$$

Difficulty

- Requires $\mathcal{O}(n^2)$ computations $\Rightarrow$ Infeasible.

Idea

- Approximate using the filtering method [Adams-2010] $\Rightarrow$ $\mathcal{O}(n)$ computations.

Current algorithms for MAP inference in dense CRFs

- Rely on the efficient filtering method [Adams-2010].

Algorithm	Time complexity per iteration	Theoretical bound
Mean Field (MF) [1]	$\mathcal{O}(n)$	No
Quadratic Programming (QP) [2]	$\mathcal{O}(n)$	Yes
Difference of Convex (DC) [2]	$\mathcal{O}(n)$	Yes
Linear Programming (LP) [2]	$\mathcal{O}(n \log(n))$	Yes (best)

Contribution

- LP in $\mathcal{O}(n)$ time per iteration
  $\Rightarrow$ An order of magnitude speedup.

[1] [Krähenbühl-2011]
[2] [Desmaison-2016]

Current algorithms for MAP inference in dense CRFs

- Rely on the efficient filtering method [Adams-2010].

Algorithm	Time complexity per iteration	Theoretical bound
Mean Field (MF) [1]	$\mathcal{O}(n)$	No
Quadratic Programming (QP) [2]	$\mathcal{O}(n)$	Yes
Difference of Convex (DC) [2]	$\mathcal{O}(n)$	Yes
Linear Programming (LP) [2]	$\mathcal{O}(n \log(n))$	Yes (best)

Contribution

- LP in $\mathcal{O}(n)$ time per iteration
  $\Rightarrow$ An order of magnitude speedup.

[1] [Krähenbühl-2011]
[2] [Desmaison-2016]

LP minimization

Current method

- Projected subgradient descent $\Rightarrow$ Too slow.
  - Linearithmic time per iteration.
  - Expensive line search.
  - Require large number of iterations.

Our algorithm

- Proximal minimization using block coordinate descent.
  - One block: Significantly smaller subproblems.
  - The other block: Efficient conditional gradient descent.
    - Linear time conditional gradient computation.
    - Optimal step size.
  - Guarantees optimality and converges faster.

LP minimization

Current method

- Projected subgradient descent $\Rightarrow$ Too slow.
  - Linearithmic time per iteration.
  - Expensive line search.
  - Require large number of iterations.

Our algorithm

- Proximal minimization using block coordinate descent.
  - One block: Significantly smaller subproblems.
  - The other block: Efficient conditional gradient descent.
    - Linear time conditional gradient computation.
    - Optimal step size.
  - Guarantees optimality and converges faster.

LP relaxation of a dense CRF

$$
\begin{aligned}
\min_{\mathbf{y}} \quad \tilde{E}(\mathbf{y}) \quad &= \sum_a \sum_i \phi_{a:i}\, y_{a:i} + \sum_{a,b \neq a} \sum_i K_{ab} \frac{|y_{a:i} - y_{b:i}|}{2}\;, \\
\text{s.t.} \quad \mathbf{y} \in \mathcal{M} &= \left\{ \; \mathbf{y} \;\middle|\; \begin{array}{l} \sum_i y_{a:i} = 1, \; a \in \{1 \ldots n\} \\ y_{a:i} \in [0,1], \; a \in \{1 \ldots n\}, \; i \in \mathcal{L} \end{array} \right\}\;,
\end{aligned}
$$

where $K_{ab} = \exp\left(\frac{-\|\mathbf{f}_a - \mathbf{f}_b\|^2}{2}\right)$.

Assumption: Label compatibility is the Potts model.

LP relaxation of a dense CRF

$$\min_{\mathbf{y}} \quad \tilde{E}(\mathbf{y}) \;=\sum_a \sum_i \phi_{a:i}\, y_{a:i} + \sum_{a,b\neq a} \sum_i K_{ab} \frac{|y_{a:i} - y_{b:i}|}{2} \;,$$

$$\text{s.t.} \quad \mathbf{y} \in \mathcal{M} = \left\{ \;\mathbf{y} \;\middle|\; \begin{array}{l} \sum_i y_{a:i} = 1, \; a \in \{1 \ldots n\} \\ y_{a:i} \in [0,1], \; a \in \{1 \ldots n\}, \; i \in \mathcal{L} \end{array} \right\} ,$$

where $K_{ab} = \exp\left(\frac{-\|\mathbf{f}_a - \mathbf{f}_b\|^2}{2}\right)$.

Assumption: Label compatibility is the Potts model.

Standard solvers would require $\mathcal{O}(n^2)$ variables.

Proximal minimization of LP

$$\begin{aligned} \min_{\mathbf{y}} \quad & \tilde{E}(\mathbf{y}) + \frac{1}{2\lambda} \left\| \mathbf{y} - \mathbf{y}^k \right\|^2 \ , \\ \text{s.t.} \quad & \mathbf{y} \in \mathcal{M} \ , \end{aligned}$$

where $\mathbf{y}^k$ is the current estimate.

Why?

- Initialization using MF or DC.
- Smooth dual $\Rightarrow$ Sophisticated optimization.

Approach

- Block coordinate descent tailored to this problem.

Proximal minimization of LP

$$\begin{aligned} \min_{\mathbf{y}} \quad & \tilde{E}(\mathbf{y}) + \frac{1}{2\lambda} \left\| \mathbf{y} - \mathbf{y}^k \right\|^2 \ , \\ \text{s.t.} \quad & \mathbf{y} \in \mathcal{M} \ , \end{aligned}$$

where $\mathbf{y}^k$ is the current estimate.

Why?

- Initialization using MF or DC.
- Smooth dual $\Rightarrow$ Sophisticated optimization.

Approach

- Block coordinate descent tailored to this problem.

Proximal minimization of LP

$$\begin{aligned} \min_{\mathbf{y}} \quad & \tilde{E}(\mathbf{y}) + \frac{1}{2\lambda} \left\| \mathbf{y} - \mathbf{y}^k \right\|^2 \ , \\ \text{s.t.} \quad & \mathbf{y} \in \mathcal{M} \ , \end{aligned}$$

where $\mathbf{y}^k$ is the current estimate.

Why?

- Initialization using MF or DC.
- Smooth dual $\Rightarrow$ Sophisticated optimization.

Approach

- Block coordinate descent tailored to this problem.

Dual of the proximal problem

$$
\begin{aligned}
\min_{\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\gamma}} g(\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\gamma}) = {} & \frac{\lambda}{2}\|A\boldsymbol{\alpha} + B\boldsymbol{\beta} + \boldsymbol{\gamma} - \boldsymbol{\phi}\|^2 \\
& + \left\langle A\boldsymbol{\alpha} + B\boldsymbol{\beta} + \boldsymbol{\gamma} - \boldsymbol{\phi}, \mathbf{y}^k \right\rangle - \langle \mathbf{1}, \boldsymbol{\beta} \rangle \ , \\
\text{s.t.} \quad & \gamma_{a:i} \geq 0 \quad \forall\, a \in \{1 \ldots n\} \quad \forall\, i \in \mathcal{L} \ , \\
& \boldsymbol{\alpha} \in \mathcal{C} = \left\{ \ \boldsymbol{\alpha} \ \middle| \ \begin{array}{l} \alpha^1_{ab:i} + \alpha^2_{ab:i} = \frac{K_{ab}}{2},\ a \neq b,\ i \in \mathcal{L} \\ \alpha^1_{ab:i}, \alpha^2_{ab:i} \geq 0,\ a \neq b,\ i \in \mathcal{L} \end{array} \ \right\} \ .
\end{aligned}
$$

Block coordinate descent

- $\boldsymbol{\alpha} \in \mathcal{C}$: Compact domain $\Rightarrow$ Conditional gradient descent.
- $\boldsymbol{\beta}$: Unconstrained $\Rightarrow$ Set derivative to zero.
- $\boldsymbol{\gamma}$: Unbounded and separable $\Rightarrow$ Small QP for each pixel.

Dual of the proximal problem

$$
\begin{aligned}
\min_{\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\gamma}} g(\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\gamma}) = \frac{\lambda}{2}\|A\boldsymbol{\alpha} + B\boldsymbol{\beta} + \boldsymbol{\gamma} - \boldsymbol{\phi}\|^2 \\
+ \left\langle A\boldsymbol{\alpha} + B\boldsymbol{\beta} + \boldsymbol{\gamma} - \boldsymbol{\phi}, \mathbf{y}^k \right\rangle - \langle \mathbf{1}, \boldsymbol{\beta} \rangle \ ,
\end{aligned}
$$

$$
\begin{aligned}
\text{s.t.} \quad & \gamma_{a:i} \geq 0 \quad \forall\, a \in \{1 \dots n\} \quad \forall\, i \in \mathcal{L} \ , \\
& \boldsymbol{\alpha} \in \mathcal{C} = \left\{ \boldsymbol{\alpha} \;\middle|\; \begin{array}{l} \alpha^1_{ab:i} + \alpha^2_{ab:i} = \frac{K_{ab}}{2},\ a \neq b,\ i \in \mathcal{L} \\ \alpha^1_{ab:i}, \alpha^2_{ab:i} \geq 0,\ a \neq b,\ i \in \mathcal{L} \end{array} \right\} \ .
\end{aligned}
$$

Block coordinate descent

- $\boldsymbol{\alpha} \in \mathcal{C}$: Compact domain $\Rightarrow$ Conditional gradient descent.
- $\boldsymbol{\beta}$: Unconstrained $\Rightarrow$ Set derivative to zero.
- $\boldsymbol{\gamma}$: Unbounded and separable $\Rightarrow$ Small QP for each pixel.

Dual of the proximal problem

$$\min_{\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\gamma}} g(\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\gamma}) = \frac{\lambda}{2}\|A\boldsymbol{\alpha} + B\boldsymbol{\beta} + \boldsymbol{\gamma} - \boldsymbol{\phi}\|^2 + \left\langle A\boldsymbol{\alpha} + B\boldsymbol{\beta} + \boldsymbol{\gamma} - \boldsymbol{\phi}, \mathbf{y}^k \right\rangle - \langle \mathbf{1}, \boldsymbol{\beta} \rangle \ ,$$

$$\text{s.t.} \quad \gamma_{a:i} \geq 0 \quad \forall\, a \in \{1 \dots n\} \quad \forall\, i \in \mathcal{L} \ ,$$

$$\boldsymbol{\alpha} \in \mathcal{C} = \left\{ \boldsymbol{\alpha} \;\middle|\; \begin{array}{l} \alpha^1_{ab:i} + \alpha^2_{ab:i} = \frac{K_{ab}}{2},\ a \neq b,\ i \in \mathcal{L} \\ \alpha^1_{ab:i}, \alpha^2_{ab:i} \geq 0,\ a \neq b,\ i \in \mathcal{L} \end{array} \right\} .$$

Block coordinate descent

- $\boldsymbol{\alpha} \in \mathcal{C}$: Compact domain $\Rightarrow$ Conditional gradient descent.
- $\boldsymbol{\beta}$: Unconstrained $\Rightarrow$ Set derivative to zero.
- $\boldsymbol{\gamma}$: Unbounded and separable $\Rightarrow$ Small QP for each pixel.

Guarantees optimality since g is strictly convex and smooth.

Conditional gradient computation

$$\forall\, a \in \{1 \dots n\}, \quad v'_a = \sum_b K_{ab}\, v_b\, \mathbb{1}[y_a \geq y_b]\ ,$$

where $y_a, y_b \in [0, 1]$.

Difficulty

- The permutohedral lattice based filtering method of [Adams-2010] cannot handle the ordering constraint.

Current method [Desmaison-2016]

- Repeated application of the original filtering method using a divide and conquer strategy $\Rightarrow \mathcal{O}(d^2 n \log(n))$ computations[1].

Our idea

- Discretize the interval $[0, 1]$ to H levels and instantiate H permutohedral lattices $\Rightarrow \mathcal{O}(Hdn)$ computations[2].

---

[1] d - filter dimension, usually 2 or 5.

[2] $H = 10$ in our case.

Conditional gradient computation

$$\forall\, a \in \{1 \dots n\}, \quad v'_a = \sum_b K_{ab}\, v_b\, \mathbb{1}[y_a \geq y_b]\ ,$$

where $y_a, y_b \in [0, 1]$.

Difficulty

- The permutohedral lattice based filtering method of [Adams-2010] cannot handle the ordering constraint.

Current method [Desmaison-2016]

- Repeated application of the original filtering method using a divide and conquer strategy $\Rightarrow \mathcal{O}(d^2 n \log(n))$ computations[1].

Our idea

- Discretize the interval $[0, 1]$ to H levels and instantiate H permutohedral lattices $\Rightarrow \mathcal{O}(Hdn)$ computations[2].

[1] d - filter dimension, usually 2 or 5.
[2] $H = 10$ in our case.

Conditional gradient computation

$$\forall\, a \in \{1 \ldots n\}, \quad v'_a = \sum_b K_{ab}\, v_b\, \mathbb{1}[y_a \geq y_b]\ ,$$

where $y_a, y_b \in [0, 1]$.

Difficulty

- The permutohedral lattice based filtering method of [Adams-2010] cannot handle the ordering constraint.

Current method [Desmaison-2016]

- Repeated application of the original filtering method using a divide and conquer strategy $\Rightarrow \mathcal{O}(d^2 n \log(n))$ computations[1].

Our idea

- Discretize the interval $[0, 1]$ to H levels and instantiate H permutohedral lattices $\Rightarrow \mathcal{O}(Hdn)$ computations[2].

[1] d - filter dimension, usually 2 or 5.

[2] $H = 10$ in our case.

Conditional gradient computation

$$\forall\, a \in \{1 \dots n\}, \quad v'_a = \sum_b K_{ab}\, v_b\, \mathbb{1}[y_a \geq y_b]\ ,$$

where $y_a, y_b \in [0, 1]$.

Difficulty

- The permutohedral lattice based filtering method of [Adams-2010] cannot handle the ordering constraint.

Current method [Desmaison-2016]

- Repeated application of the original filtering method using a divide and conquer strategy $\Rightarrow \mathcal{O}(d^2 n \log(n))$ computations[1].

Our idea

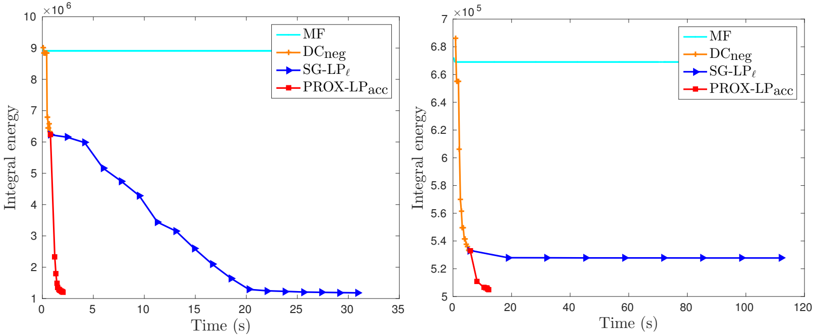
- Discretize the interval $[0, 1]$ to H levels and instantiate H permutohedral lattices $\Rightarrow \mathcal{O}(Hdn)$ computations[2].

---

[1] d - filter dimension, usually 2 or 5.

[2] $H = 10$ in our case.

Segmentation results



*Assignment energy as a function of time for an image in (**left**) MSRC and (**right**) Pascal.*

- Both the LP minimization algorithms are initialized with $\mathrm{DC}_{\mathrm{neg}}$.

Segmentation results

		Ave. E ($\times 10^3$)	Ave. T (s)	Acc.
MSRC	MF5	8078.0	0.2	79.33
	MF	8062.4	0.5	79.35
	DC_{neg}	3539.6	1.3	83.01
	$SG\text{-}LP_{\ell}$	3335.6	13.6	83.15
	$PROX\text{-}LP_{acc}$	1340.0	3.7	84.16
Pascal	MF5	1220.8	0.8	79.13
	MF	1220.8	0.7	79.13
	DC_{neg}	629.5	3.7	80.43
	$SG\text{-}LP_{\ell}$	617.1	84.4	80.49
	$PROX\text{-}LP_{acc}$	507.7	14.7	80.58

Results on the MSRC and Pascal datasets.

Segmentation results

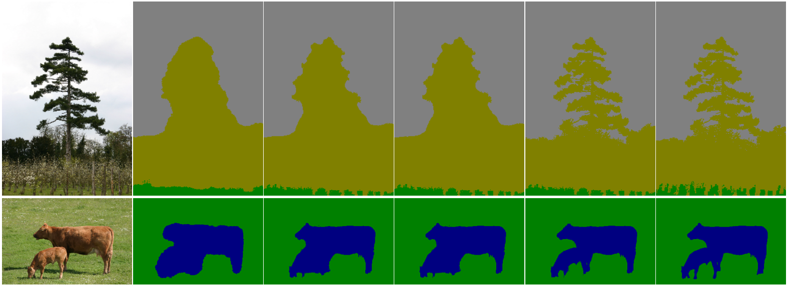
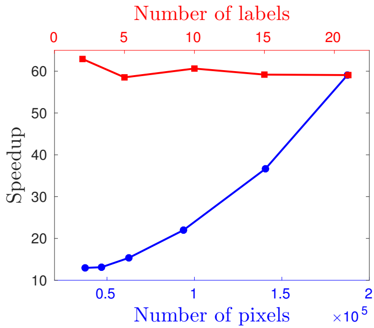


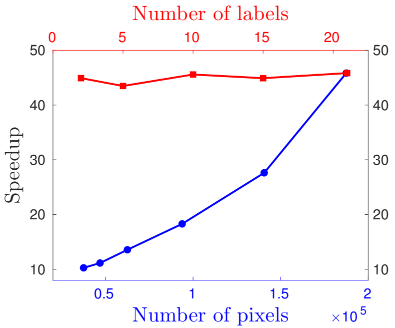
Image MF $\mathrm{DC}_{\mathrm{neg}}$ $\mathrm{SG\text{-}LP}_{\ell}$ $\mathrm{PROX\text{-}LP}_{\mathrm{acc}}$ Ground truth

Qualitative results on MSRC.

Modified filtering algorithm



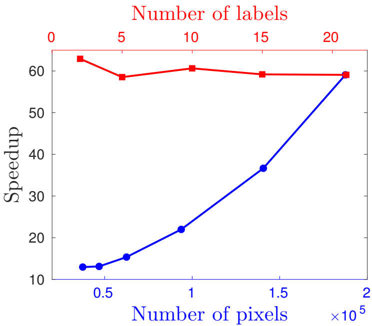
Spatial kernel ($d = 2$)



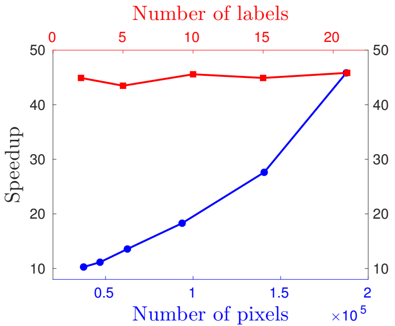
Bilateral kernel ($d = 5$)

Speedup of our modified filtering algorithm over the divide and conquer strategy on a Pascal image.

Modified filtering algorithm



Spatial kernel ($d = 2$)



Bilateral kernel ($d = 5$)

Speedup of our modified filtering algorithm over the divide and conquer strategy on a Pascal image.

Speedup is around $45 - 65$ on the standard image.

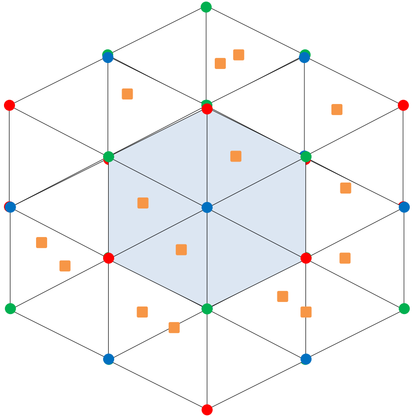
Conclusion

- We have introduced the first LP minimization algorithm for dense CRFs whose iterations are linear in the number of pixels and labels.

Arxiv: `https://arxiv.org/abs/1611.09718`

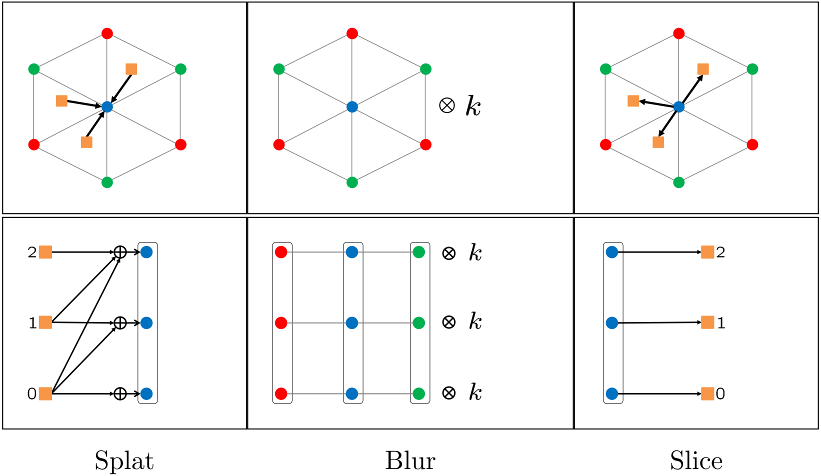
Thank you!

Permutohedral lattice



A 2-dimensional hyperplane tessellated by the permutohedral lattice.

Modified filtering algorithm



Top row: *Original filtering method.* **Bottom row:** *Our modified filtering method.* $H = 3$.