CSE 659A: Advances in Computer Vision

Spring 2019: T-R: 2:30-4pm @ Cupples II/230
Instructor: Ayan Chakrabarti (ayan@wustl.edu).
http://www.cse.wustl.edu/~ayan/courses/cse659a/

Feb 7, 2019

GENERAL

- Homework reviews due Today 11:59 PM! (No extensions)
- Next paper to review will be posted by tonight.
- Remember that the first project is due in a month!
  - Start thinking about what paper you’ll want to implement.
  - I’ll ask you to submit a brief proposal at the end of next week, listing the paper and a para about what you’ll do.

STEREO


- Use matching cost + SGM to get an initial disparity map $d^0[x, y]$.
- Use SLIC to get an initial set of superpixels $\{P_i\}$


- Use matching cost + SGM to get an initial disparity map $d^0[x, y]$.
- Use SLIC to get an initial set of superpixels $\{P_i\}$
- Define an MRF model over these superpixels

\[
\log P(y_i) = \ldots
\]

- For each super-pixel, we’ll have a node in our graph corresponding to a variable $y_i$.
  $y_i = [\alpha_i, \beta_i, \gamma_i]$ corresponds to the plane parameters for superpixel $i$. We’ll be solving for $y_i$s, not disparities, because we assume each super-pixel is planar.

- Include a unary energy for each $y_i$ based on consistency with $d^0[x]$: 

\[
\phi_i(y_i) = \sum_{(x,y) \in P_i} \|d^0[x, y] - \alpha_i x - \beta_i y - \gamma_i\|
\]

Robust Version:

\[
\phi_i(y_i) = \min \left( \sum_{(x,y) \in P_i} \|d^0[x, y] - \alpha_i x - \beta_i y - \gamma_i\|, K \right)
\]

- Use matching cost + SGM to get an initial disparity map $d^I[x, y]$.
- Use SLIC to get an initial set of superpixels $\{P_i\}$
- Define an MRF model over these superpixels

$$\log P(\{y_i\}) = -\sum_i \phi_i(y_i)\ldots, y_i \in \mathbb{R}^3$$

- For each super-pixel, we’ll have a node in our graph corresponding to a variable $y_i$ and energy $\phi_i(y_i)$.
- Now consider two super-pixels $P_i$ and $P_j$ which share a boundary.
- Introduce a discrete variable $o_{ij}$ with four possible labels for the relationship between the two super-pixels: Co-planar (co), Hinge (h), i occludes j (lo), j occludes i (ro).
- Connect $y_i, y_j, o_{ij}$ to each other, and include a three-way clique energy $\theta_{ij}(o_{ij}, y_i, y_j)$.

Also add a constant term which says hinge is less preferable than co-planar.

This energy will include penalties for whether the assignments to $y_i, y_j$ supports the label $o_{ij}$.

- Use matching cost + SGM to get an initial disparity map \( d^0(x, y) \).
- Use SLIC to get an initial set of superpixels \( \{ P_i \} \)
- Define an MRF model over these superpixels

\[
\log P(y_i, \{ o_{ij} \}) = - \sum_i \phi_i(y_i) - \sum_{ij} \theta_{ij}(o_{ij}, y_i, y_j) \quad y_i \in \mathbb{R}^3, \; o_{ij} \in \{ co, h, lo, ro \}
\]

if \( o_{ij} = lo \): \( \theta_{ij}(o_{ij}, y_i, y_j) = \theta_{ij}^o(y_i, y_j) + \lambda_{occ} \)
if \( o_{ij} = ro \): \( \theta_{ij}(o_{ij}, y_i, y_j) = \theta_{ij}^o(y_i, y_j) + \lambda_{occ} \)

\[
\theta_{ij}^o(y_{front}, y_{back}) = \begin{cases} 
\lambda_{mp} & \text{if } \exists (x, y) \in B_{ij} \text{ s.t. } \hat{d}(x, y; y_{front}) < \hat{d}(x, y; y_{back}) \\
0 & \text{otherwise}
\end{cases}
\]

Almost done, but Yamaguchi also adds terms that considers groups of three and four superpixels that all touch each other.

It includes 3 and 4-way clique potentials for these, but defined only on the discrete labels \( o_{ij} \). Adds a penalty when the set of labels represent an impossible configuration.

- Use matching cost + SGM to get an initial disparity map $d^0[x, y]$.
- Use SLIC to get an initial set of superpixels $\{P_i\}$
- Define an MRF model over these superpixels

$$\log P(y_i, \{o_j\}) = - \sum_i \phi_i(y_i) - \sum_{ij} \theta_j(o_{ij}, y_i, y_j) - \sum_{ijkl} \Phi(o_{ij}, o_{ik}, o_{il}, o_{jk})$$

This gives us a complex graph! Lots of different kinds of clique potentials. Both continuous and discrete variables.

- Convert to pairwise factor graph.
- Use particle belief propagation.
- Gave state-of-the-art results at the time.

Remember SLIC: Get super-pixels by assigning a label to each pixel:

$$\sum_{n=1}^{K} \text{arg min}_{\alpha \in \{1, \ldots, K\}} \sum_{n=1}^{K} \left( \| I[n] - \mu_k \|^2 + \beta \| n - \mu_k \|^2 \right)$$

Solved this as K-means clustering in intensity + position space.

But these superpixels are based only on appearance. We have the initial disparity map $d^0[x, y]$. Why not use that information as well?
Use matching cost + SGM to get an initial disparity map \( d^0[x, y] \).
Use SLIC to get an initial set of superpixels \( \{ P_i \} \)

Remember SLIC: Get super-pixels by assigning a label \( L[n] \in \{ 1, \ldots, K \} \) to each pixel \( n \):

\[
L = \arg \min_{L[n]} \sum_{k=1}^{K} \sum_{L[n]=k} \alpha \| I[n] - \mu_k \|^2 + \beta \| n - \mu_k \|^2
\]

Simple Version

- Just add a term for disparity similarity.

\[
L = \arg \min_{L[n]} \sum_{k=1}^{K} \sum_{L[n]=k} \alpha \| I[n] - \mu_k \|^2 + \beta \| n - \mu_k \|^2 + \gamma \| d^0[n] - \mu_k \|^2
\]

But this groups together pixels with similar disparity values.

What we want is a method that groups together pixels if they lie on a plane.

I.e., we want to impose a planar model, not a "fronto-planar" model.

Stereo SLIC

\[
L = \arg \min_{L[n]} \sum_{k=1}^{K} \sum_{L[n]=k} \alpha \| I[n] - \mu_k \|^2 + \beta \| n - \mu_k \|^2 + \gamma \| d^0[n] - \mu_k \|^2
\]

Next in addition to the color and spatial centroids, we also have plane parameters \( \alpha_k, \beta_k, \gamma_k \) for each superpixel.

How do we minimize?

- Same alternating minimization.
- Given centroids and plane parameters: at each pixel figure out which \( k \) minimizes the cost.
- Given labels, compute the centroids as means of all pixels assigned to that label ...

and \( \alpha_k, \beta_k, \gamma_k \) by fitting a plane to disparity of all pixels currently labeled \( k \).

Works surprisingly well! Gives us much-better superpixels (and can use fewer / larger superpixels).

- In fact, if you now just fit the resulting superpixels to planes (simpler no-MRF), it does a really good job.
- Defining the MRF on these super-pixels now does even better.

The MRF model is defined on overlapping cliques. Slow.
Use matching cost + SGM to get an initial disparity map $d^0[x, y]$. 
1. Use SLIC to get an initial set of superpixels $\{P_i\}$ 
2. Use an MRF model to further smooth the results with planarity models.

Fundamentally, why do we break this into two steps?

- SLIC is based on finding a "partition" of pixels. Fast. Use on all pixels.
- The MRF model is defined on overlapping cliques. Slow. Use on reduced number of superpixels.

So we are using the efficient SLIC method to reduce the number of variables, so that we can do MRF energy minimization.

Consider a dense set of overlapping patches at multiple scales ...
... of sizes $4 \times 4$, $8 \times 8$, $16 \times 16$, $32 \times 32$, $64 \times 64$ with stride 1!

So, we aren't doing a partition, or "picking" a scale.

Now define an MRF:

$$
\log P(d, \{I_j\}) = - \sum_j \Theta(I_j, \{d[n]\})_{n \in P_i}
$$

If $I_j = 0$ and $P_i$ is not a plane (for example, it has a discontinuity or a hinge),

$$
\Theta(I_j = 0, \{d[n]\}) = \tau_i
$$

This is a constant (doesn't depend on $d$), but we can pick a different constant for each patch. Do this based on variance of left-image intensities inside the patch.
Chakrabarti et al., "Low-level Vision by Consensus in a Spatial Hierarchy of Regions," CVPR 2015

- Use matching cost + SGM to get an initial disparity map $d^0[x, y]$.
- Define an energy on overlapping cliques of pixels.

Consider a dense set $\{P_i\}$ of overlapping patches at multiple scales, and introduce a binary variable $I_i$ that encodes whether the patch is a plane (1) or not (0).

Now define an MRF:

$$\log P(d, \{I_i\}) = -\sum_i \Theta_i(I_i, \{d[n]\}_{n \in P_i})$$

If $I_i = 1$ and $P_i$ is a plane:

$$\Theta_i(I_i = 0, \{d[n]\}) = \min_{\theta} \sum_{n \in P_i} (\theta^T[n, 1] - d^0[n])^2 + \lambda(\theta^T[n, 1] - d[n])^2$$

$\theta = [\alpha, \beta, \gamma]^T$, so $\theta^T[x, y, 1] = \alpha x + \beta y + \gamma$.

What we’re saying is that all $d[n]$ in that patch should be explained by a planar model that also matches the initial disparity map $d^0[n]$.

---

Chakrabarti et al., "Low-level Vision by Consensus in a Spatial Hierarchy of Regions," CVPR 2015

$\downarrow$

$$d[n], \{I_i\}, \{\theta_i\} = \arg\min_{I_i = 0} \sum_i \tau_i + \sum_{i, j = 1} \sum_{n \in P_i} (\theta_i^T[n, 1] - d^0[n])^2 + \lambda(\theta_i^T[n, 1] - d[n])^2$$

- Create plane variables $\theta_i$ for each patch.
- Alternate between updating
  - $\{I_i, \theta_i\}$ with $d[n]$ fixed.
  - $d[n]$ with $\{I_i, \theta_i\}$ fixed.

(In practice, we also begin with a smaller value of $\lambda$ and increase it across iterations.)

---

Chakrabarti et al., "Low-level Vision by Consensus in a Spatial Hierarchy of Regions," CVPR 2015

$\downarrow$

$$d[n], \{I_i\}, \{\theta_i\} = \arg\min_{I_i = 0} \sum_i \tau_i + \sum_{i, j = 1} \sum_{n \in P_i} (\theta_i^T[n, 1] - d^0[n])^2 + \lambda(\theta_i^T[n, 1] - d[n])^2$$

- Choose $I_i = 0$ and $\theta_i = 0$ for a patch when either the initial disparity $d^0[n]$, or the estimated disparity $d[n]$ (which depends on other patches), can’t be well explained as a plane.
- This is still based on overlapping cliques. So, a value of $d[n]$ will have to agree with predictions of its values from all patches that include it and are labeled planar.
- Instead of saying each pixel belongs to one superpixel (to be determined), and every superpixel is planar (decided), we’re saying each pixel belongs to a number of overlapping patches (decided) but only a subset of patches are planar (to be determined).
- How do we minimize this?

Alternating iterations!
Chakrabarti et al., "Low-level Vision by Consensus in a Spatial Hierarchy of Regions," CVPR 2015

\[ d[n], \{ I_i \}, \{ \theta \} = \arg \min_{i, j = 0} \sum_{i, j = 0} \tau_i + \sum_{i, j = 0} \sum_{n \in P_i} \left( \theta_i^2[n, 1] - d^2[n] \right)^2 + \lambda (\theta_i^2[n, 1] - d[n])^2 \]

- Alternate between updating
  - \{ I_i, \theta \} with \{ d[n] \} fixed.
  - \{ d[n] \} with \{ I_i, \theta \} fixed.

For each pixel independently,

\[ d[n] = \frac{\sum_{i \in \mathcal{P}} I_i \theta_i^2[n, 1]}{\sum_{i \in \mathcal{P}} I_i} \]

That is, the average of \{ d[n] \}'s predictions from all patches that include it and labeled planar.

Chakrabarti et al., "Low-level Vision by Consensus in a Spatial Hierarchy of Regions," CVPR 2015

\[ d[n], \{ I_i \}, \{ \theta \} = \arg \min_{i, j = 0} \sum_{i, j = 0} \tau_i + \sum_{i, j = 0} \sum_{n \in P_i} \left( \theta_i^2[n, 1] - d^2[n] \right)^2 + \lambda (\theta_i^2[n, 1] - d[n])^2 \]

- Alternate between updating
  - \{ I_i, \theta \} with \{ d[n] \} fixed.
  - \{ d[n] \} with \{ I_i, \theta \} fixed.

So we can use alternating minimization, and the computation is independent per-patch and per-pixel.

But there’s still a lot of patches!

We’re considering \( 8 \times 8 \ldots 64 \times 64 \) with stride 1. Number of patches at each scale \( \approx \) number of pixels, and we have 4 scales.

Use the fact that we have regular patches. Convert things to convolutions!

Represent everything as image like objects:

- \{ d[n] \} is \( H \times W \times 1 \).
- For each scale, of all patches \( S \times S \):
  - \{ I_i \} as \( I^S \) which is \( (H - S + 1) \times (W - S + 1) \times 1 \)
  - \{ \theta \} as \( \theta^S \) which is \( (H - S + 1) \times (W - S + 1) \times 3 \)

\( I^S[n] \) and \( \theta^S[n] \) correspond to parameters for the \( S \times S \) patch with top-left pixel at \( n \).

\( I^S[n] \) and \( \theta^S[n] \) are:

\[ \sum_{i \in \mathcal{P}} I_i \theta_i^2[n, 1] \]

\[ \sum_{i \in \mathcal{P}} I_i \]

\[ \sum_{i \in \mathcal{P}} I_i \theta_i^2[n, 1] \]

\[ \sum_{i \in \mathcal{P}} I_i \]

Which patches at a specific scale \( S \) include \( n \)?

Those that correspond to locations \( (n - S + 1), \ldots, n \) in \( I^S, \theta^S \).
Chakrabarti et al., "Low-level Vision by Consensus in a Spatial Hierarchy of Regions," CVPR 2015

\[ d[n], \{I_l\}, \{\theta_l\} = \arg\min_{l \geq 0} \sum_{i,l=1} t_l + \sum_{i,l=1} (\theta_i^T [n, 1] - d^0[n])^2 + \lambda(\theta_i^T [n, 1] - d[n])^2 \]

- Alternate between updating
  - \{I_l, \theta_l\} with \{d[n]\} fixed.
  - \{d[n]\} with \{I_l, \theta_l\} fixed.

\[ d[n] = \theta[n]^2 T [n, 1], \quad \theta[n] = \frac{\alpha[n]}{\beta[n]} \]

\[ \alpha[n] = \sum_{i \in \mathcal{P}, l} I_l \theta_l = \sum_i \text{Conv}(I^S \theta^S), \text{ones}(S, S), \text{‘full’}) \]
\[ \beta[n] = \sum_{i \in \mathcal{P}, l} I_l = \sum_i \text{Conv}(I^S), \text{ones}(S, S), \text{‘full’}) \]

But wait, there's more!

\[ \text{ones}(S, S) = \text{Conv}(\text{ones}(S/2, S/2), \text{ones}(2, 2)^T, \text{‘full’}) \]
\[ \left[ \begin{array}{cccc}
1 & 1 & 1 & 1 \\
1 & 1 & 0 & 0 \\
1 & 0 & 1 & 0 \\
1 & 0 & 0 & 0 
\end{array} \right] * \left[ \begin{array}{c}
1 \\
0 \\
1 \\
1 
\end{array} \right] \]

We can use recursive computation (break S into \( \log_2 S \) 2x2 convs), and use the fact that we also have scale S/2 in our set of patches.

---

Chakrabarti et al., "Low-level Vision by Consensus in a Spatial Hierarchy of Regions," CVPR 2015

\[ d[n], \{I_l\}, \{\theta_l\} = \arg\min_{l \geq 0} \sum_{i,l=1} t_l + \sum_{i,l=1} (\theta_i^T [n, 1] - d^0[n])^2 + \lambda(\theta_i^T [n, 1] - d[n])^2 \]

- Alternate between updating
  - \{I_l, \theta_l\} with \{d[n]\} fixed.
  - \{d[n]\} with \{I_l, \theta_l\} fixed.

\[ \theta_i = \arg\min_{n \in \mathcal{P}} \sum_n (\theta_i^T [n, 1] - d^0[n])^2 + \lambda(\theta_i^T [n, 1] - d[n])^2 \]
\[ \theta_i = \arg\min_{n \in \mathcal{P}} \theta_i^T \mathcal{Q} \theta_i - 2 \theta_i^T b_i \]
\[ \mathcal{Q}_i = (1 + \lambda) \sum_{n \in \mathcal{P}} \left[ \begin{array}{c} n \\ 1 \end{array} \right] [n, 1] \]

- 3 \times 3 matrix. Doesn't depend on \{d[n]\} so doesn't change with iterations.
Chakrabarti et al., "Low-level Vision by Consensus in a Spatial Hierarchy of Regions," CVPR 2015

\[
d[n], \{I_i\}, \{\theta_i\} = \arg\min \sum_{i:j=0} \tau_i + \sum_{i:j=1} \sum_{\theta \in \mathcal{P}} (\theta_i^T \{n, 1\} - d[n])^2 + \lambda (\theta_i^T \{n, 1\} - d[n])^2
\]

- Alternate between updating
  - \{I_i, \theta_i\} with \(d[n]\) fixed.
  - \(d[n]\) with \{I_i, \theta_i\} fixed.

\[
\theta_i = \arg\min_{\theta \in \mathcal{P}_i} \sum_{n \in \mathcal{P}_i} (\theta_i^T \{n, 1\} - d[n])^2 + \lambda (\theta_i^T \{n, 1\} - d[n])^2 = \arg\min \theta_i^T Q \theta_i - 2 \theta_i^T b_i
\]

\[
b_i = \sum_{n \in \mathcal{P}_i} (d[n] + \lambda d[n]) \left[ \begin{array}{c} n \\ 1 \end{array} \right] = \sum_{n \in \mathcal{P}_i} D[n]
\]

- \(b_i^{S_i} \{n\} = \text{Conv}(b_i^{S_i}, \text{ones}(2,2); 'x', 'valid'))