HW Implementation of MRF MAP Inference on an FPGA Platform

Jungwook Choi and Rob A. Rutenbar

FPL 2012

Aug. 30 2012

Overview

- Goal: Accurate & fast HW MRF MAP solver
  - Why MRF MAP inference and HW impl.?
  - Loopy belief propagation
  - Tree-reweighted message passing (TRW-S)
  - Our TRW-S hardware architecture
  - FPGA experimental results (x30 faster than SW)
  - Conclusion & future work

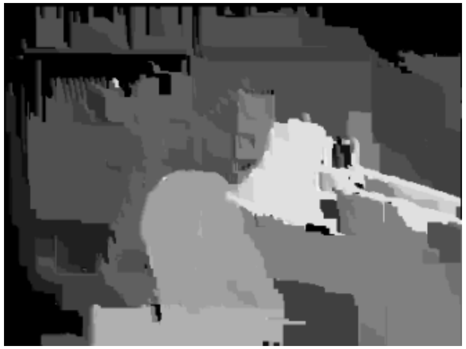
MRF MAP Inference

Maximum a posteriori (MAP)

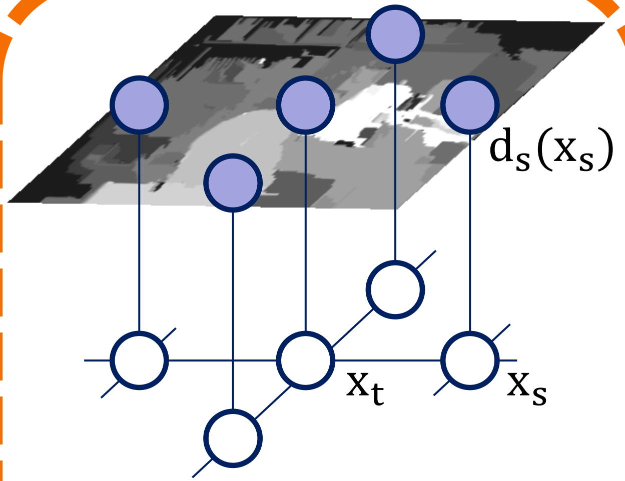
$$\underset{\mathbf{x}}{\operatorname{argmax}}\, \underbrace{P(\overset{\text{Label assignments}}{\mathbf{x}}|\overset{\text{Observations}}{\mathbf{y}})}_{\text{Posterior}} = \underset{\mathbf{x}}{\operatorname{argmax}}\, \underbrace{P(\mathbf{y}|\mathbf{x})}_{\text{Likelihood}}\,\underbrace{P(\mathbf{x})}_{\text{Prior}}$$

Energy minimization on Markov random fields (MRF)

$$\underset{\mathbf{x}}{\operatorname{argmin}} \left(\qquad \text{Energy}\ (\mathbf{x}) \qquad \right)$$

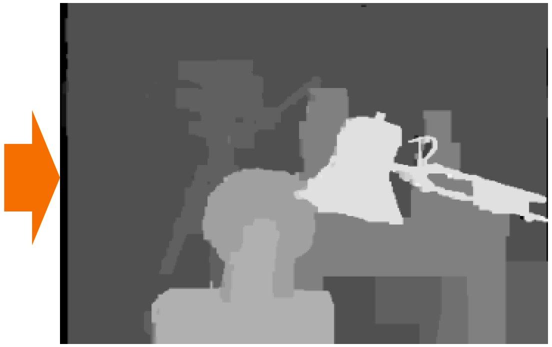


3D depth map by matching pixels along the line



Likelihood → Data cost

Prior → Smoothness cost



3D depth map by MRF MAP inference

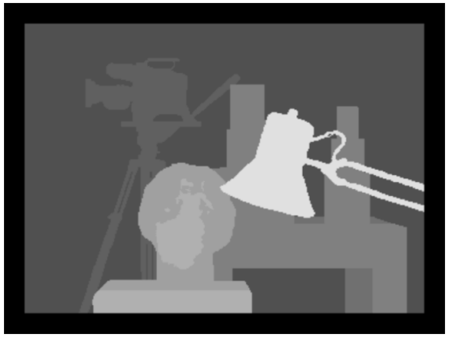
MRF MAP Inference

Maximum a posteriori (MAP)

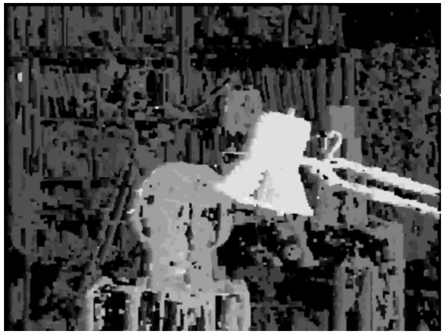
$$\underset{\mathbf{x}}{\operatorname{argmax}}\, \underbrace{\mathrm{P}(\overset{\text{Label assignments}}{\mathbf{x}}|\overset{\text{Observations}}{\mathbf{y}})}_{\text{Posterior}} = \underset{\mathbf{x}}{\operatorname{argmax}}\, \underbrace{\mathrm{P}(\mathbf{y}|\mathbf{x})}_{\text{Likelihood}}\underbrace{\mathrm{P}(\mathbf{x})}_{\text{Prior}}$$

Energy minimization on Markov random fields (MRF)

$$\underset{\mathbf{x}}{\operatorname{argmin}} \left(\text{Energy}\ (\mathbf{x}) \right)$$

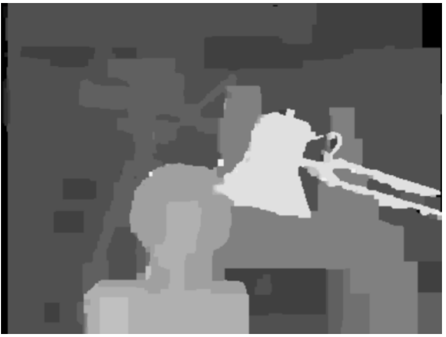


Ground truth

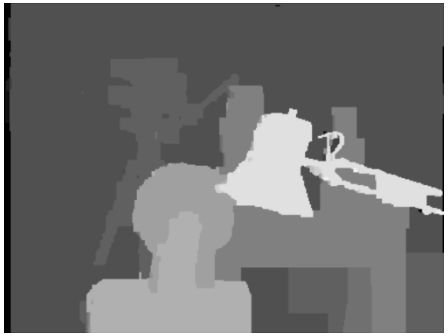


Greedy method

(Iterated conditional modes)



Belief propagation(BP)



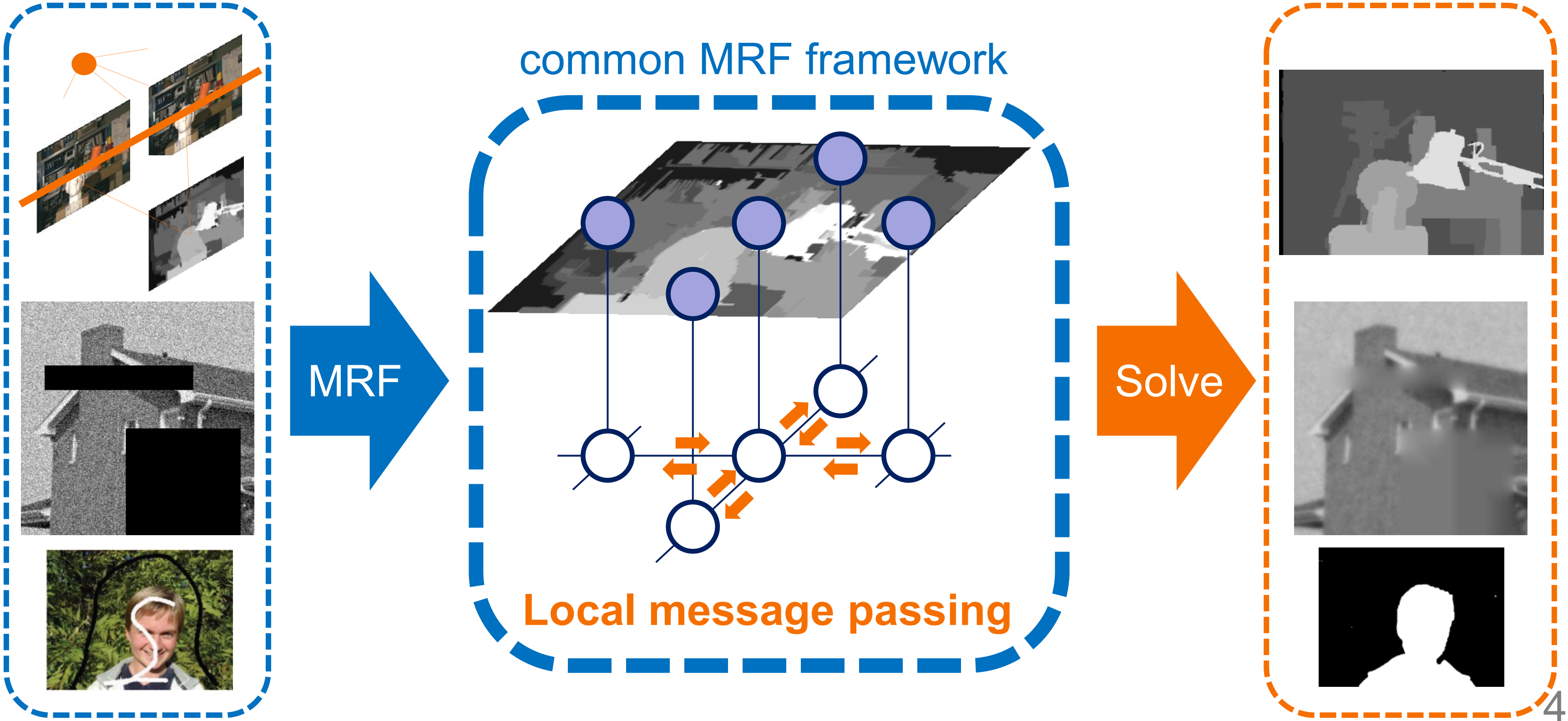
Tree-reweighted(TRW)

Energy minimization on MRF

Images from http://vision.middlebury.edu/MRF/results/tsukuba/

Why Custom Hardware Impl.?

- Many apps map to a *common* MRF framework
- Computation is *local*, well matched for custom HW

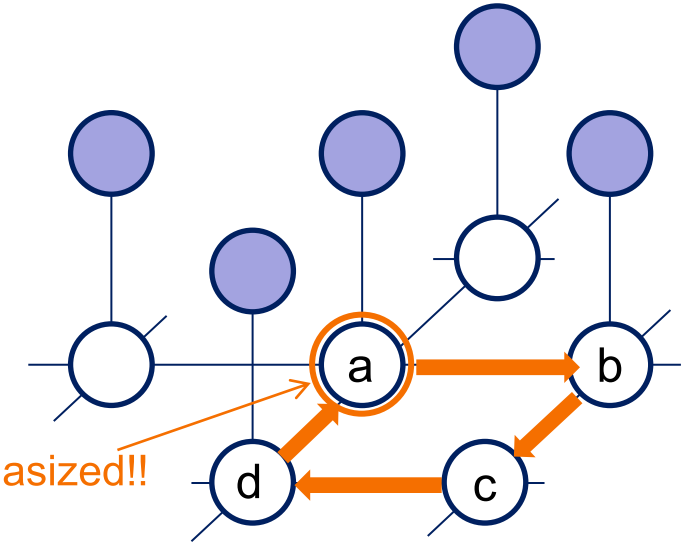


common MRF framework

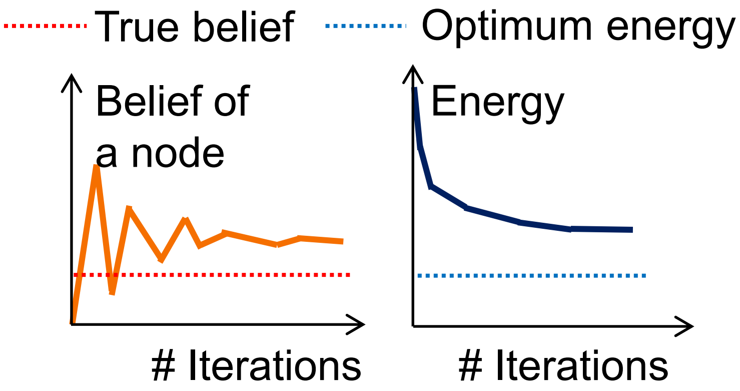
Local message passing

Loopy Belief Propagation

- In **BP**, a node propagates *belief* to neighbors by passing *messages*
  - Message: "based on what I know now, what do I tell to my neighbor?"
  - Belief: "what label should I choose based on my neighbors?"
  - Energy computed by the best labels based on beliefs
- BP on a tree
  - Optimum energy can be found after all inward/outward *message passing* is done
- BP on a loopy graph
  - No guarantee of optimality due to *loops*

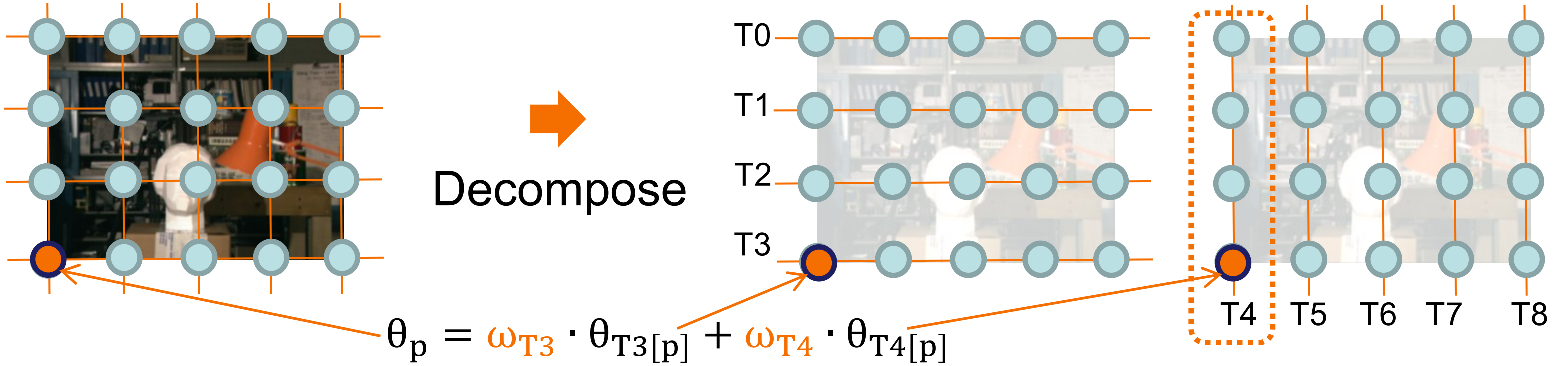


Overemphasized!!



Tree-Reweighted Message Passing

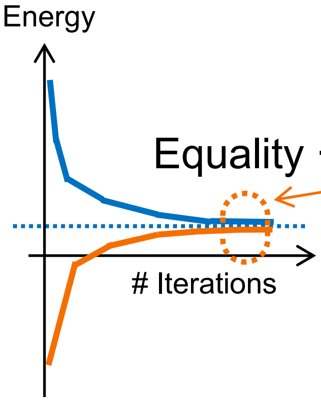
- Idea: decompose a loopy graph to a set of *trees*



$$\theta_p = \omega_{T3} \cdot \theta_{T3[p]} + \omega_{T4} \cdot \theta_{T4[p]}$$

  - Energy is the weighted sum of tree energy

$$\text{Energy}(\mathbf{x}) = \sum_{\text{trees T}} \underbrace{\omega_T}_{\text{Tree weight}} \cdot \underbrace{\text{Energy}_T(\mathbf{x_T})}_{\text{Energy over a tree}}$$



Equality → optimum energy!

$$\underbrace{\min_{\mathbf{x}} \text{Energy}(\mathbf{x})}_{\text{Goal: minimize overall energy}} \geq \underbrace{\sum_{\text{trees T}} \omega_T \cdot \left(\min_{\mathbf{x_T}} \text{Energy}_T(\mathbf{x_T})\right)}_{\text{New goal: maximize this lower bound}}$$

BP on a tree
➔ No loop!!

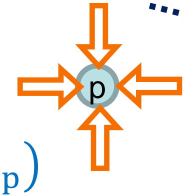
Sequential TRW (TRW-S)

- New goal: maximize *lower bound* by

  data cost update & message passing on trees

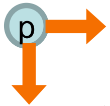
- Sequential message passing ➔ convergence property
  - *Lower bound* is guaranteed not to decrease
  - ➔ More chance to find the optimum energy!!

- Data cost update

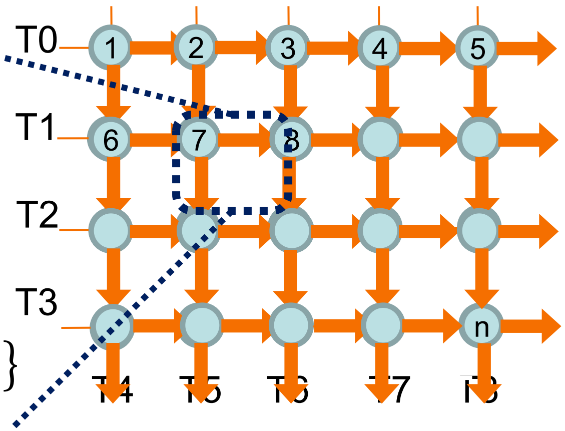


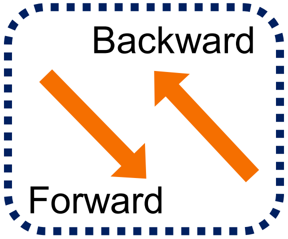
$$\hat{\theta}_p(x_p) = d_p(x_p) + \sum_{s\in Nb(p)} M_{sp}(x_p)$$

- Message passing



$$M_{pq}(x_q) = \min_{x_p}\{(\gamma_{pq}\cdot\hat{\theta}_p(x_p) - M_{qp}(x_p)) + V_{pq}(x_p, x_q)\}$$



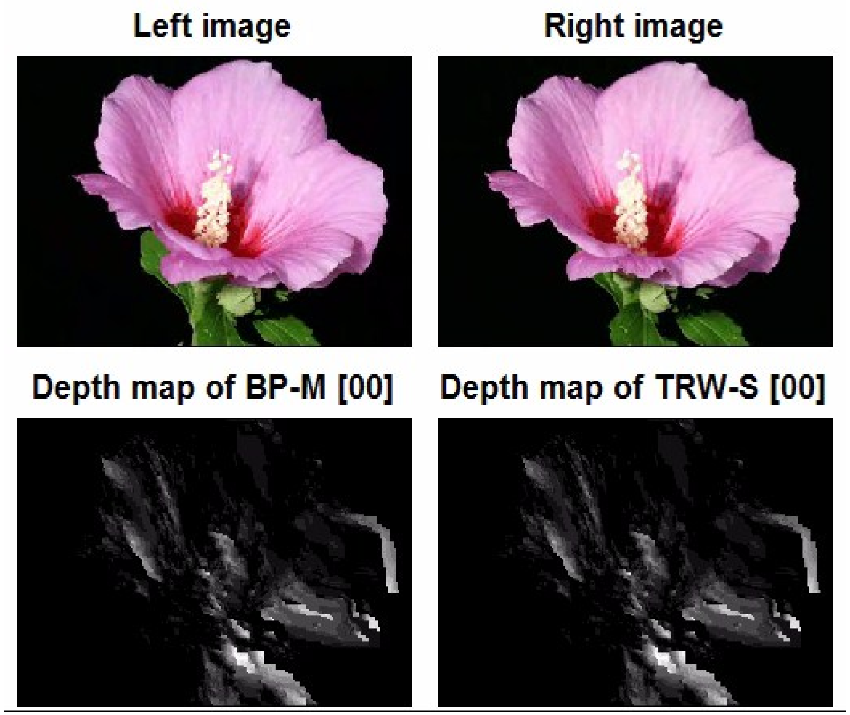


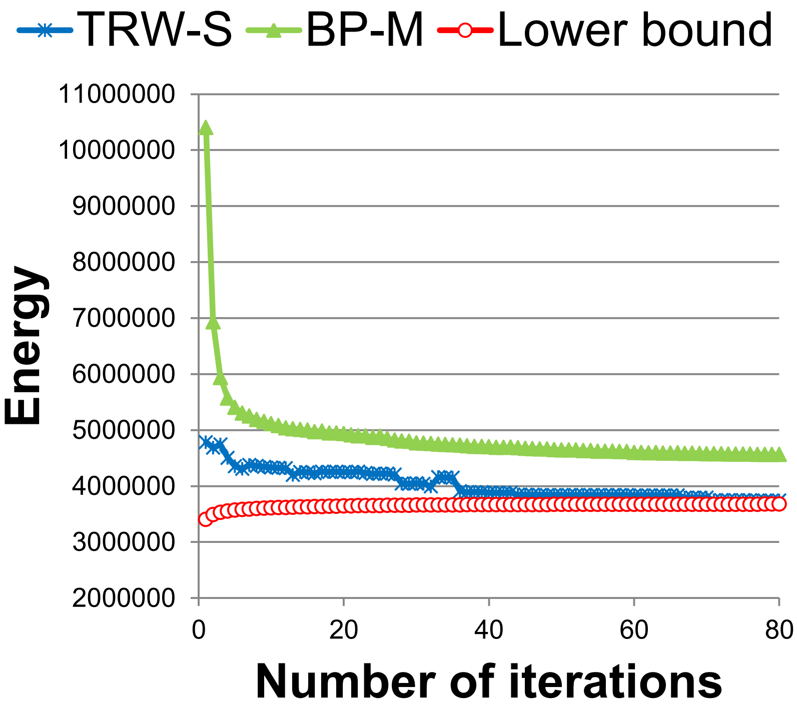
1 iteration

- Challenge : parallelize “sequential message passing”

Comparison: BP-M and TRW-S

- Benchmark: Flower stereo images* (360x262x16 label)
  - BP-M: min-sum belief propagation, run 80 iterations.
  - TRW-S: sequential tree reweighted message passing, run 80 iterations.

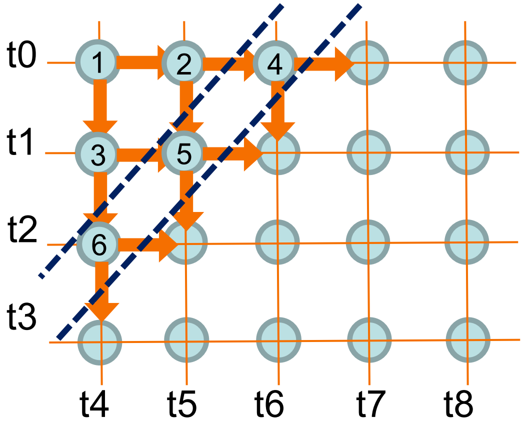


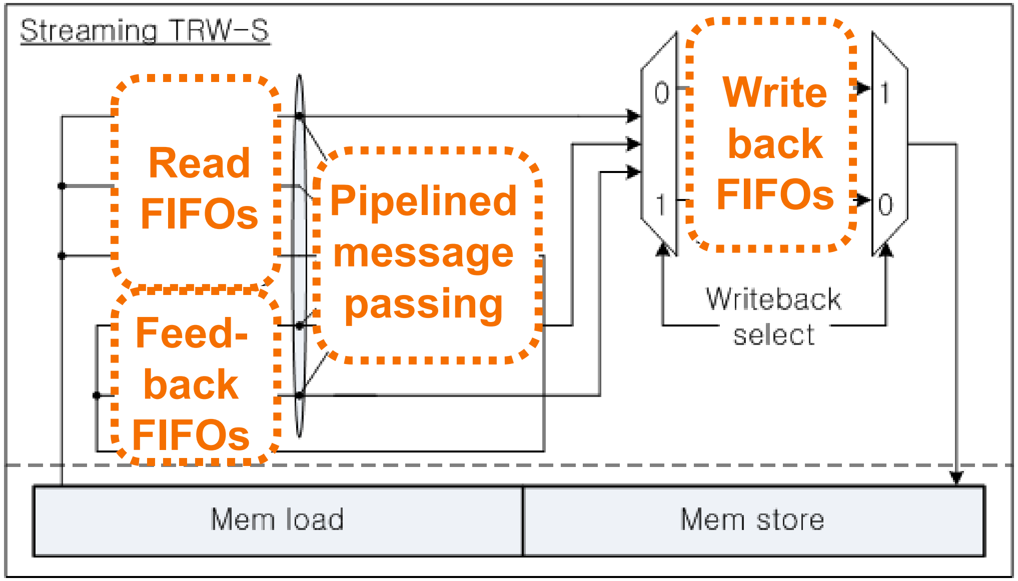


*From stereo movie sample, http://www.stereomaker.net/sample/index.html

Streaming TRW-S HW Architecture

- Key: *diagonal ordering* of message passing for *parallelism*
- Decoupled, streaming arch.
- Launch/retire 1 pixel/clock
  - Complete label-set likelihood updates for all labels
- Deep pixel-proc pipeline
  - 14 stages deep
  - So: 14 pixels “in flight” / clock

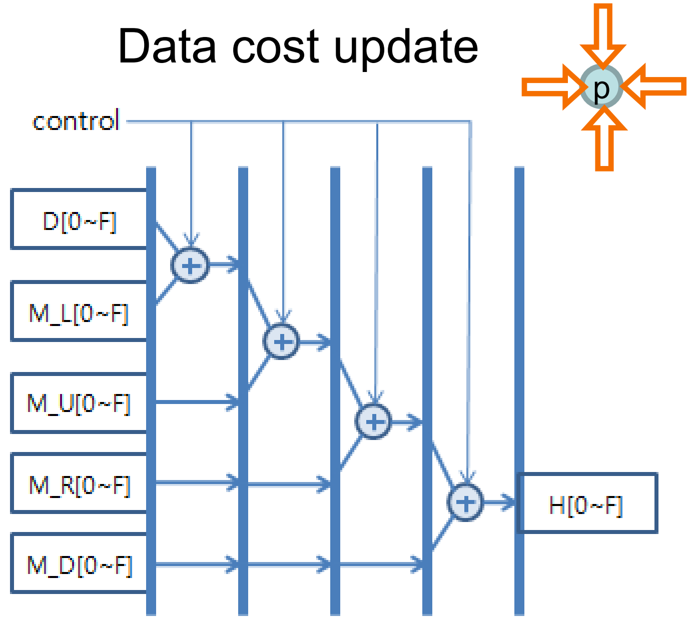




Streaming TRW-S HW Architecture

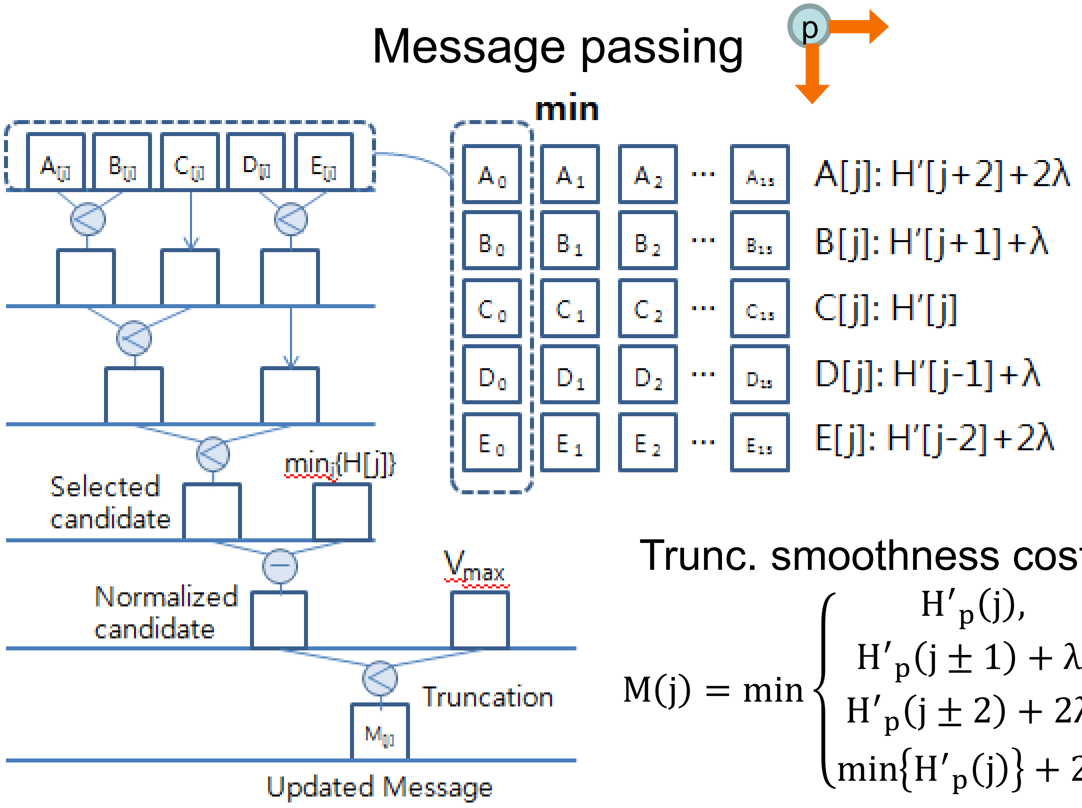
– Pipelined message passing

Data cost update



$$H_p(x_p) = d_p(x_p) + \sum_{s \in Nb(p)} M_{sp}(x_p)$$

Message passing

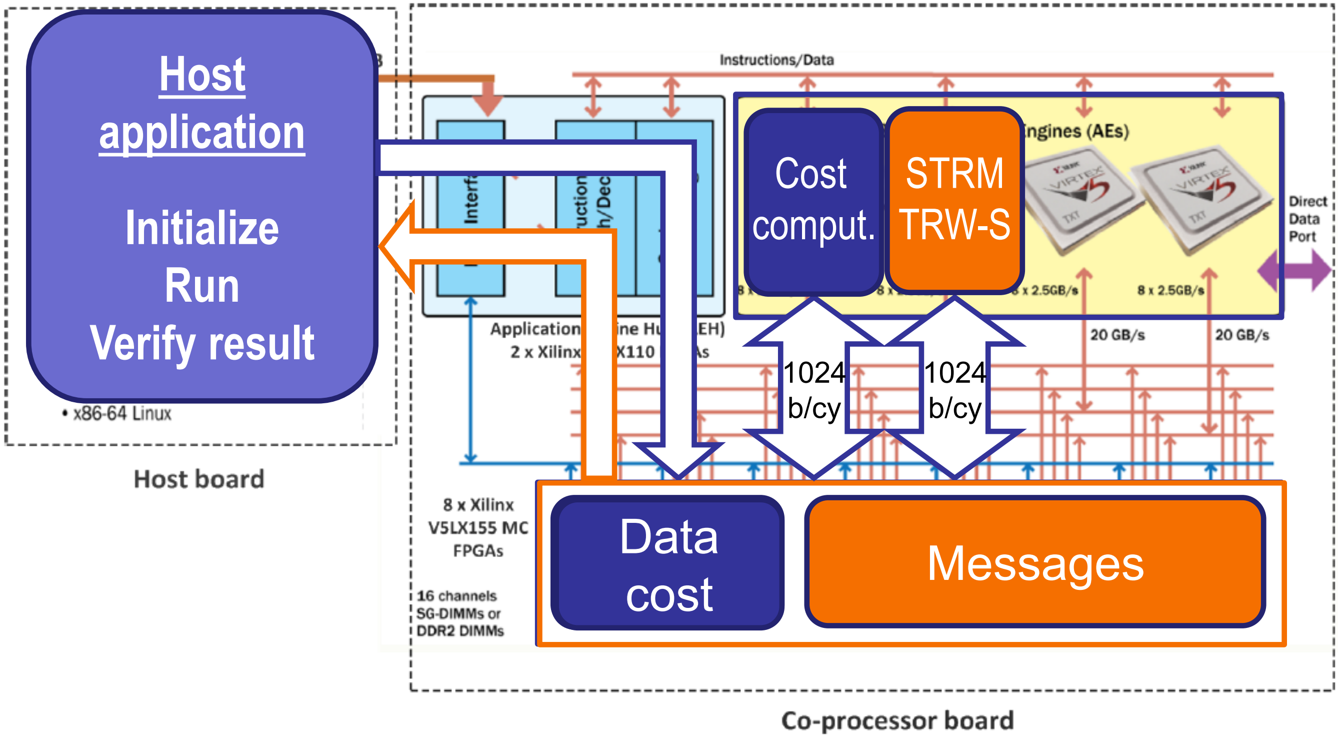


Trunc. smoothness cost

$$M(j) = \min \begin{Bmatrix} H'_p(j), \\ H'_p(j \pm 1) + \lambda, \\ H'_p(j \pm 2) + 2\lambda, \\ \min\{H'_p(j)\} + 2\lambda \end{Bmatrix}$$

Experimental Platform: FPGA

- Our platform: Convey HC-1
  - Host-FPGA cache-coherent virtual memory system
  - Max memory BW: 1Kbit/cycle(~20GB/sec)/FPGA (runs @150MHz)



Experimental Results

- Stereo matching of Middlebury benchmark*
  - Speed (per iteration)
    - FPGA impl. of streaming TRW-S (F-sTRW-S) runs in Convey HC-1 (@ 150MHz)
    - SW impl. [Szeliski 2008] runs in Intel Core i7 (@ 1.87GHz)

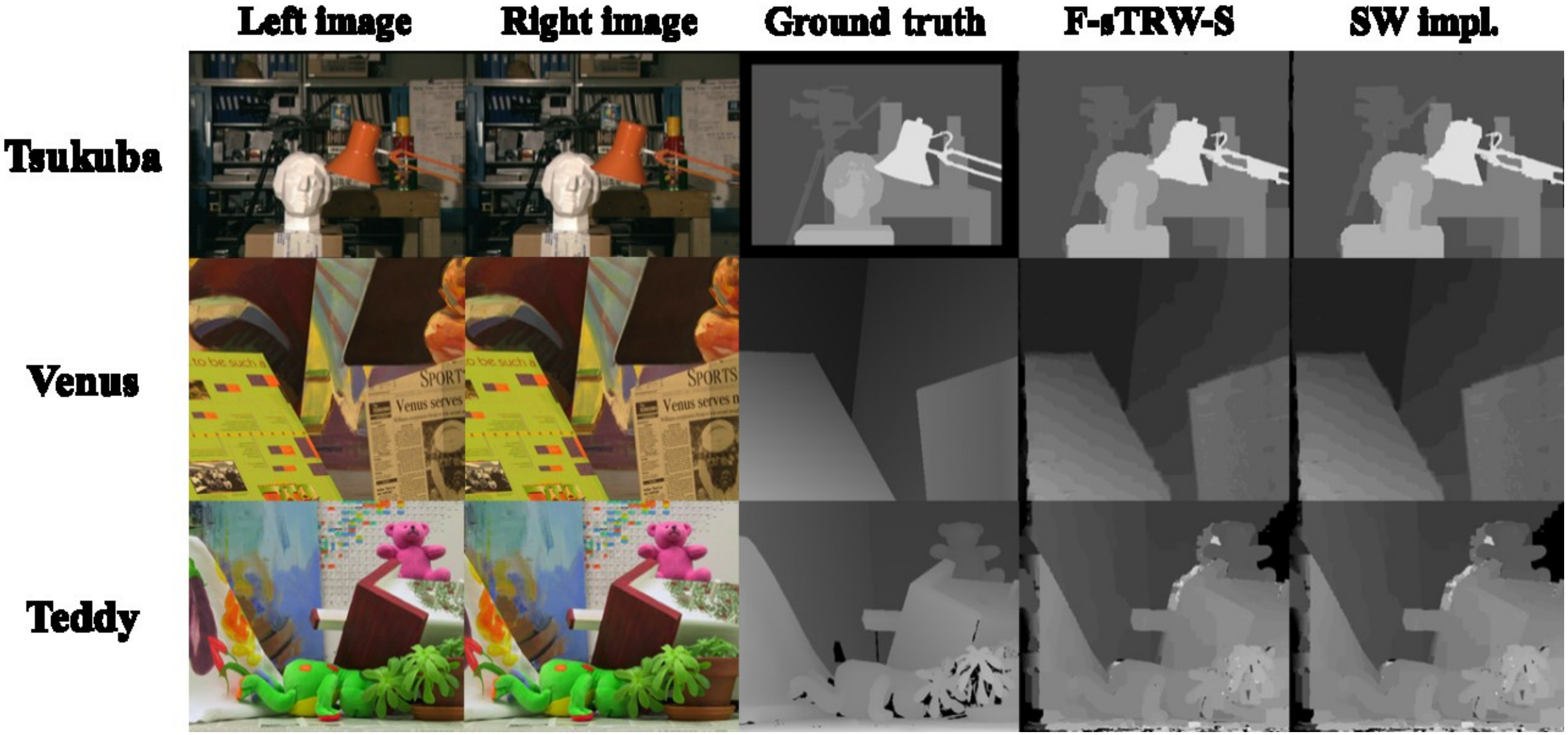
Task	Task Size	Cost Fn.	Our HW: F-sTRW-S		SW Impl.*
Tsukuba	384x288x16L	Truncated linear	478,134 cy	**0.0032 sec**	0.12 sec
Venus	434x383x20L	Truncated quadratic	1,436,257 cy	**0.0096 sec**	0.47 sec
Teddy	450x375x60L	Potts model	2,914,599 cy	**0.0194 sec**	0.67 sec

- F-sTRW-S is 34.5~49.0 times faster than SW impl.

*R. Szeliski, et al., "A Comparative Study of Energy Minimization Methods for Markov Random Fields with Smoothness-Based Priors," *IEEE Tr. PAMI*, 2008..

Experimental Results

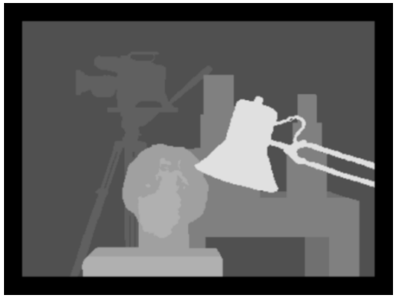
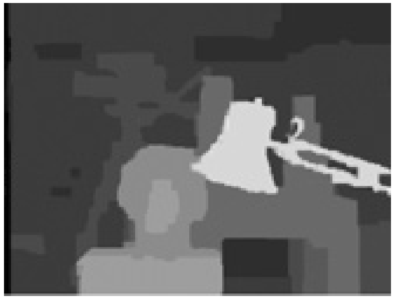
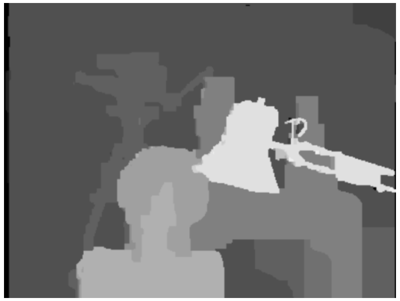
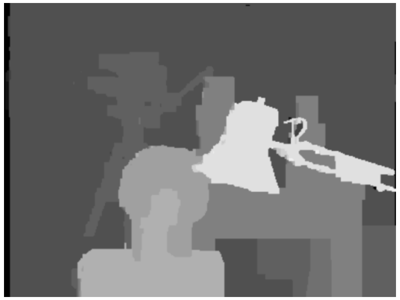
- Stereo matching of Middlebury benchmark (cont'd)
  - Comparison of 3D depth maps after 500 iterations



  - F-sTRW-S speeds-up SW impl without loss of quality of results

Experimental Results

- Rough comparison with other VLSI impl. [Liang 2011]

Algorithm	**Tile-based BP***	**F-sTRW-S**	
Spec.	320x240x64L	384x288x16L (max: 512x512x64L)	
			
Num. of Iteration	$(B, T_I, T_O) = (16, 20, 5)$	$T_O = 5$	$T_O = 40$
Minimum Energy	396,953	**393,434**	**370,359**
Speed	7.28 frames/sec	**38.32 frames/sec**	**7.25 frames/sec**

  - F-sTRW-S shows compelling speed and inference capability

*Liang, et al., "Hardware-Efficient Belief Propagation," *IEEE Trans. Circ. Syst. Video Tech*, May 2011.

Experimental Results

- Comparison of speed with other GPU impl.

Impl.	Real-time BP* [Yang 2006]	Tile-based BP** [Liang 2011]	Fast BP*** [Xiang 2012]	F-sTRW-S
GPU	NVIDIA GeForce 7900 GTX	NVIDIA GeForce 8800 GTS	NVIDIA GeForce GTX 260	N/A
# Iteration	(4 coarse to fine scales) = (5,5,10,20)	$(B, T_I, T_O) = (16, 20, 5)$	(3 coarse to fine scale) = (9,6,2)	$T_O = 5$
Time (ms)	79.71	124.38	61.41	**26.10**

  – F-sTRW-S outperforms other GPU impl. in speed.

* Q. Yang, et al., “Real-time global stereo matching using hierarchical belief propagation,” *BMVC,* 2006.
** Liang, et al., “Hardware-Efficient Belief Propagation,” *IEEE Trans. Circ. Syst. Video Tech*, May 2011.
*** X. Xiang, et al., “Real-time stereo matching based on fast belief propagation,” *MACH VISION APPL*, 2012

Conclusion & Future work

- Conclusion
  - The FIRST custom hardware implementation of Sequential tree-reweighted message passing (TRW-S) algorithm is introduced.
  - Our streaming TRW-S implementation shows not only compelling speed but also superior quality of results compared to other belief propagation implementation on VLSI and GPU.
- Future work
  - Streaming video-rate TRW-S stereo matching engine
  - Expand Streaming TRW-S for more apps

Key References

- R. Szeliski, et al., “A Comparative Study of Energy Minimization Methods for Markov Random Fields with Smoothness-Based Priors,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 30, no. 6, pp. 1068-1680, Jun. 2008.
- J. Sun, et al., “Stereo Matching Using Belief Propagation,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 25, no. 7, pp. 787-800, Jul. 2003.
- V. Kolmogorov, “Convergent Tree-Reweighted Message Passing for Energy Minimization,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 28, no. 10, pp. 1568-1583, Oct. 2006.
- Convey computer, “Convey HC-1 Personality Development Kit Reference Manual, v 4.1,” http://www.conveycomputer.com, Sep. 2009.
- C. -K. Liang, et al., “Hardware-Efficient Belief Propagation,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 21, no. 5, pp. 525-537, May 2011.
- Q. Yang, et al., “Real-time global stereo matching using hierarchical belief propagation,” *The British Machine Vision Conference*, pp. 989-998, 2006.
- X. Xiang, et al., “Real-time stereo matching based on fast belief propagation,” *Machine Vision and Applications*, pp. 1-9, 2012.

Thank You