# EFFICIENT EDGE PRESERVING STEREO MATCHING

*Cevahir Çığla*†,* and *A. Aydın Alatan*†
† Department of Electrical and Electronics Engineering, M.E.T.U, Turkey
* VESTEK Electronic Research & Development Corp
e-mail: cevahir@eee.metu.edu.tr, alatan@eee.metu.edu.tr

## ABSTRACT

A computationally efficient stereo matching algorithm is introduced providing high precision dense disparity maps via local aggregation approach. The proposed algorithm exploits a novel paradigm, namely separable successive weighted summation (SWS) among horizontal and vertical directions with constant operational complexity, providing effective connected 2D support regions based on local color similarities. The intensity adaptive aggregation enables crisp disparity maps which preserve object boundaries and depth discontinuities. The same procedure is also utilized to diffuse information through overlapped pixels during occlusion handling. According to the experimental results on Middlebury online stereo benchmark, the proposed method is one of the most effective local stereo algorithm providing high quality disparity models by unifying constant time filtering and weighted aggregation. Hence, the proposed algorithm provides a competitive alternative, with its efficient GPU and FPGA implementations, for various local methods in terms of achieving precise disparity maps from stereo video within fast execution time.

## 1. INTRODUCTION

- **Two main** constraints of stereo matching:
  - Computational complexity
  - High quality
- **Local** methods:
  - Exploit windowed support regions and aggregation
  - Enables faster processing compared to global methods
- ***Adaptive weights***:
  - Provide crisp and high quality disparity maps
  - However, require huge computation

## 2. CONTRIBUTIONS

- Extends two pass integral filtering to **weighted support**
  - O(1) complexity
  - **Fastest** aggregation
  - **Connected** support regions
  - Efficient memory access
  - High quality matching
- Occlusion handling
  - **Local background** is favored
  - Color and depth weighting are unified
  - Constant time depth diffusion

## 3. ALGORITHM

- The conventional steps of local methods is followed
  1. Cost Calculation
  2. **Cost Aggregation → novelty**
  3. Minimization
  4. **Occlusion handling → novelty**

### 3.1 Cost Calculation

- Sum of Absolute Difference and Hamming distance over Census Transform are combined.

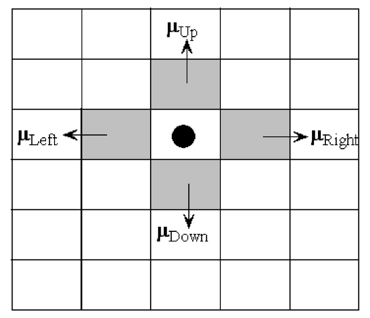
$$C_d^{SAD}(x,y)=\min\left(\sum_{i=1}^{3}\left|I_{left}(x,y,i)-I_{Right}(x+d,y,i)\right|,T\right)$$

$$C_d^{CENSUS}(x,y)=Ham(CT_{left}(x,y),CT_{Right}(x+d,y))$$

$$C_d(x,y)=\alpha.C_d^{SAD}(x,y)+(1-\alpha)C_d^{CENSUS}(x,y)$$

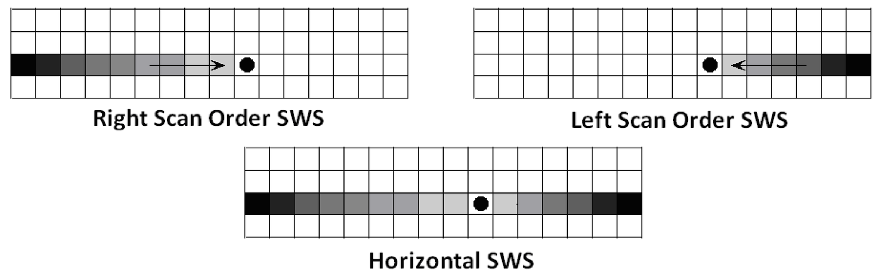
### 3.2 Aggregation

- Involves **three** stages; **Stage-1**: permeability weights of each pixel are calculated in four directions:




$$\mu=\min(e^{(-\Delta R/\sigma)},e^{(-\Delta G/\sigma)},e^{(-\Delta B/\sigma)})$$

- **Stage-2**: Horizontal aggregation is achieved by successive weighted summation (**SWS**) in left-right and right-left directions




$$C_d^R(x)=C_d(x)+\mu_R(x-1).C_d^R(x-1)$$

Update rule in left-right direction

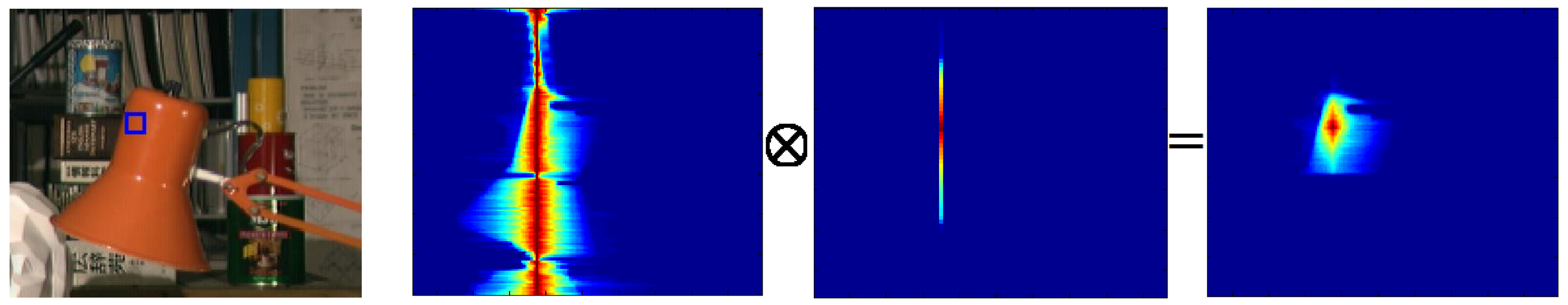
Explicit Form

$$C_d^R(x)=C_d(x)+\sum_{i=1}^{x-1}\{C_d(x-i).\underbrace{\prod_{j=1}^{i}\mu_R(x-j)}_{W_{eff}^R(x-i)}\}$$

$$=C_d(x)+\sum_{i=1}^{x-1}C_d(x-i).W_{eff}^R(x-i)$$

Effective weight

- **Stage-3**: SWS in vertical direction over horizontally aggregated data.




Hor. Aggr. Ver. Aggr. Effective 2D Support Region

- Proposed method utilizes
  **6 additions** and **4 multiplications** per pixel.

### 3.3 Minimization

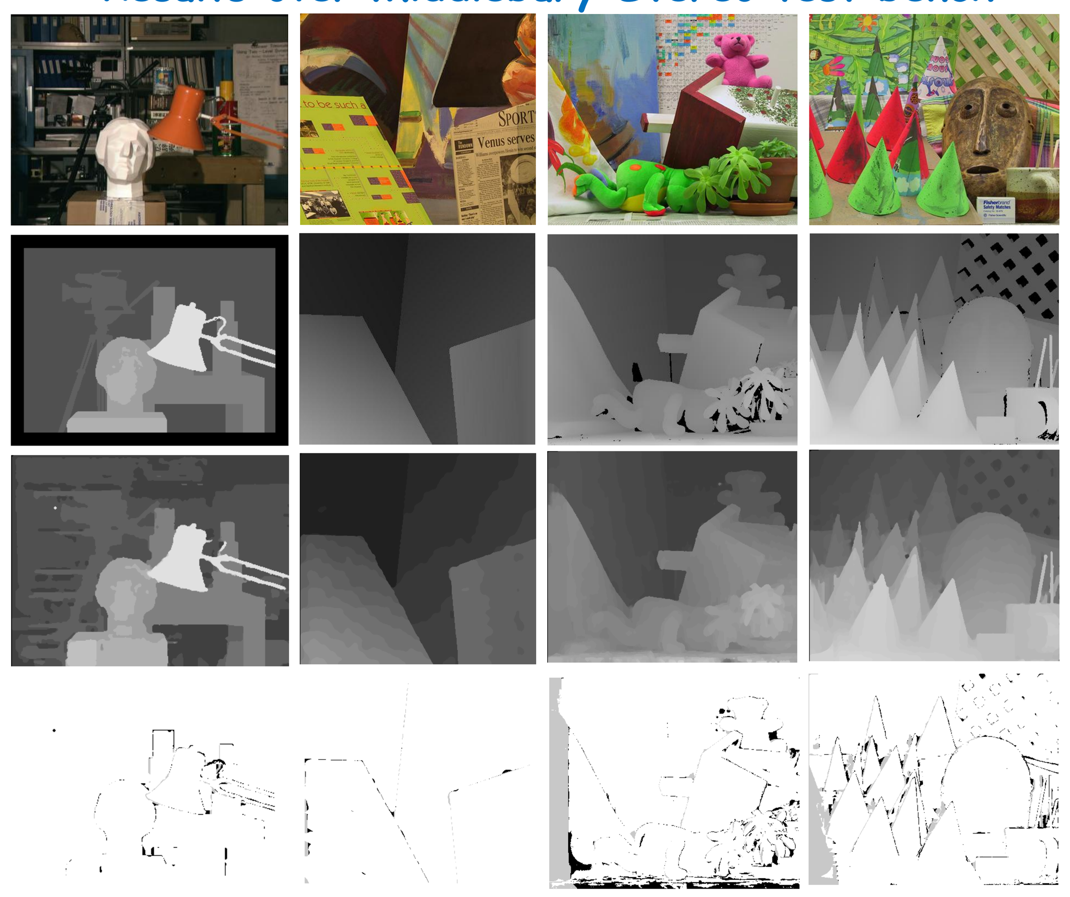
- **Winner Take All** optimization is performed among disparity candidates

### 3.4 Occlusion Handling

- The unreliable pixels are detected by left-right cross check.
- A Confidence map in constructed
  - "**0**" for occluded pixels
  - **Lower** weight to **local foreground**
- Disparity values are scaled with confidence and filtered by **SWS**

## 4. EXPERIMENTAL RESULTS

Results over Middlebury Stereo test bench



Computation Times for Different Platforms

| | FPGA | GPU | CPU |
|---|---|---|---|
| Image Resolution | 480x270 | 360x288 | 360x288 |
| Disparity Candidate | 30 | 30 | 30 |
| Computation speed | 60 fps | 25 fps | 3 fps |

Rankings of the Selected Local Methods

| Algorithm | Rank | Avg. Error [%] | Error non-occluded pixels [%] Tsukuba | Venus | Teddy | Cones |
|---|---|---|---|---|---|---|
| **Proposed (SAD+Census)** | 12 | ***5.50*** | ***1.06*** | 0.32 | ***5.60*** | ***2.65*** |
| CostFilter [2] | 15 | 5.55 | 1.51 | 0.20 | 6.16 | 2.71 |
| GeoSup | 18 | 5.80 | 1.45 | ***0.14*** | 6.88 | 2.94 |
| AdaptDisp | 23 | 6.10 | 1.19 | 0.23 | 7.80 | 3.62 |
| DistinctSM | 29 | 6.14 | 1.21 | 0.35 | 7.45 | 3.91 |
| **Proposed (SAD)** | 33 | 6.33 | 1.06 | 1.00 | 5.86 | 4.06 |
| SegSupport | 37 | 6.44 | 1.25 | 0.25 | 8.43 | 3.77 |
| CostAggOcc | 38 | 6.20 | 1.38 | 0.44 | 6.80 | 3.60 |
| AdaptWeight | 44 | 6.67 | 1.38 | 0.71 | 7.88 | 3.97 |
| VarCross [1] | 56 | 7.60 | 1.99 | 0.62 | 9.75 | 6.28 |

CPU Time Comparison of Cost Aggregation

| seconds | Tsukuba | Venus | Teddy | Cones |
|---|---|---|---|---|
| Image Resolution | 384x288 | 434x383 | 450x375 | 450x375 |
| Disparity Candidate | 16 | 20 | 60 | 60 |
| VarCross [1] | **0.109** | 0.192 | 0.616 | 0.623 |
| Guided Filter [2] | 2.494 | 4.626 | 13.877 | 13.891 |
| Proposed | 0,128 | **0,174** | **0,501** | **0,575** |

- Algorithm ranks **12** and **1** among all and local methods respectively in Middlebury bench.
- The **fastest** aggregation in **CPU** among top 30

- Regular and successive data access patern enabled implementation in
  - **GPU (*Nvidia GeForce GTX 480*)**
  - **FPGA (*Spartan 6 slx 45 nm*)**

FPGA

A shot from TV

**References**
[1] Ke Zhang, Jiangbo Lu and Gauthier Lafruit, **Cross based stereo matching using orthogonal integral images**, IEEE TCSVT, 2009
[2] C. Rhemann, A. Hosni, M. Bleyer, C. Rother, and M. Gelautz, **Fast cost-volume filtering for visual correspondence and beyond**, CVPR 2011.
[3] K. He, J. Sun, and X. Tang. **Guided Image Filtering**, In European Conference on Computer Vision, 2010.

@ICCV LDRMC Workshop 2011