embedded VISION SUMMIT

Efficiently Computing Disparity Maps for Low-Cost 3D Stereo Vision

cognivue CORPORATION

Tom Wilson, CogniVue Corporation

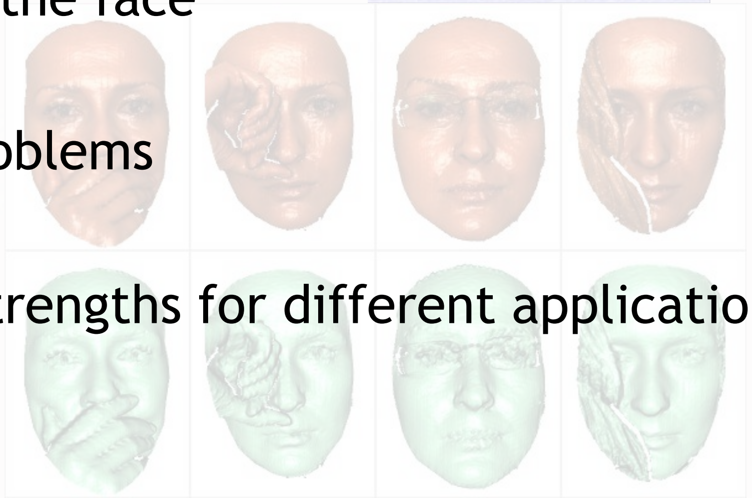
October 2nd, 2013

Outline

- Reasons for 3D sensing
- Different approaches for 3D sensing and trade-offs
  - Use cases, power, cost
- Stereo vision and disparity mapping
  - Dense vs. Sparse
- Dense disparity mapping: algorithmic approaches
  - Simple aggregation and computational load estimate
  - Results and next step
- Architectural example

Motivation for 3D Sensing

- 2D Computer Vision has fundamental challenges with:
  - Segmentation: foreground from background
  - Illumination: e.g. with face recognition
  - Relative position: places objects in the scene
  - Occlusion: e.g. hands in front of the face



- 3D sensor depth map solves these problems

- Different 3D sensors with different strengths for different application requirements

Comparing 3D Sensing Technologies

- Stereo has advantages in range and low power sensor (although does require lighting unlike the other approaches)
- High computational complexity is required to derive high quality depth map

Adapted from Calaço et al 2013

Technology	Frame Rate	Daylight Sensitivity	Depth Resolution	Total Power	Working Range
Leap	90-100 fps	high	1 mm	3 - 5 W	< 0.6 m
Ultrasound	50 fps	none	coarse ~ 2-5 cm	~300 mW	< 1 m
Stereo Camera	25-30 fps	none	coarse > 5 cm	200 mW*	0 - ~100 m
Structured Light	30 fps	high	coarse > 2 cm	3-5 W	0.8-3 m
Time of Flight	30 fps	high	< 1 cm	3-5 W	0-2 m

Comparing 3D Sensing Technologies

- Stereo has advantages in range and low power sensor (although does require lighting unlike the other approaches)
- High computational complexity is required to derive high quality depth map

Technology	Frame Rate	Daylight Sensitivity	Depth Resolution	Total Power	Working Range
Leap	90-100 fps	high	1 mm	3 - 5 W	< 0.6 m
Ultrasound	50 fps	none	coarse ~ 2-5 cm	~300 mW	< 1 m
Stereo Camera	25-30 fps	none	coarse > 5 cm	200 mW*	0 - ~100 m
Structured Light	30 fps	high	coarse > 2 cm	3-5 W	0.8-3 m
Time of Flight	30 fps	high	< 1 cm	3-5 W	0-2 m

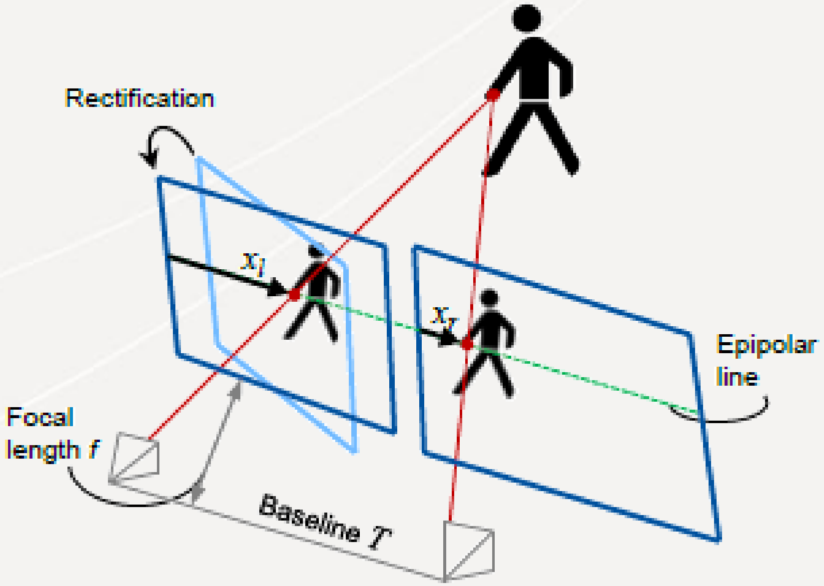
Adapted from Calaço et al 2013

Subaru EyeSight™

- Range of 80m
- Completely vision-based active safety system for Adaptive Cruise Control and Automatic Emergency Braking

Stereo Matching

- Dx and Dy are the disparities for objects x and y respectively in the left and right image frames
- Larger disparities imply closer objects (so Dx > Dy)
- Calculate a disparity map and this corresponds to a 3D depth map (e.g. disparities are assigned gray-scale values)



Local vs. Global

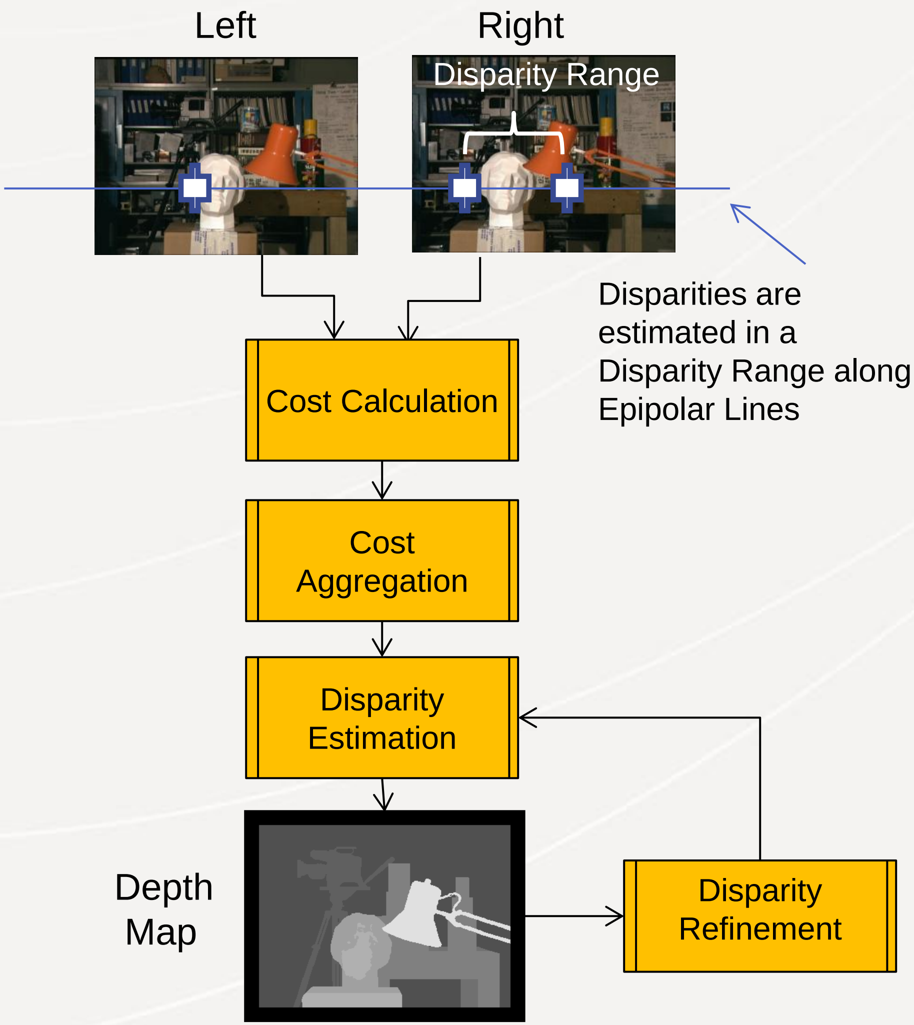
Most stereo algorithms can be placed one of two categories

- Local—the disparity calculation is dependent on intensity values in set windows in the stereo images.
- Global—stereo matching problem converted to global function; goal to optimize this global function that combines matching cost and smoothness cost terms
- Local is generally preferred for embedded implementations although Global tends to perform better in Middlebury test evaluation

Sparse vs. Dense Disparity Mapping

- Sparse: Disparity calculated for features (using FAST, SIFT, SURF, etc.) in left image vs. right image along a Disparity Range
- Dense: Disparity calculated for every pixel in the left image frame vs. pixels in the right image along a range of possible disparities using a “cost” calculation (e.g. SAD, SSD, etc.)
- Dense disparity in general produces more reliable results using Middlebury dataset

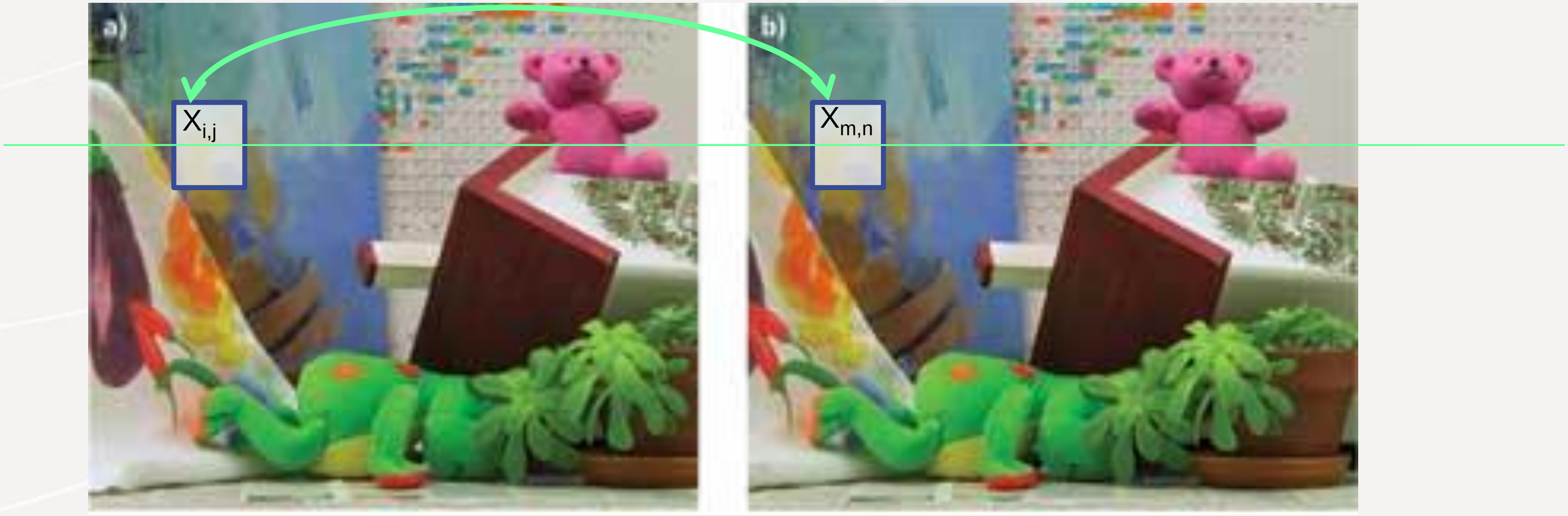
Disparity Mapping Process



- Assume images are "rectified" before cost calculations
- Cost calculation: pixel in left image is correlated to pixels in right image in a disparity range
- Cost aggregation: drives decision on disparity level with the best match (lowest cost)
- Disparity Refinement to discard errors

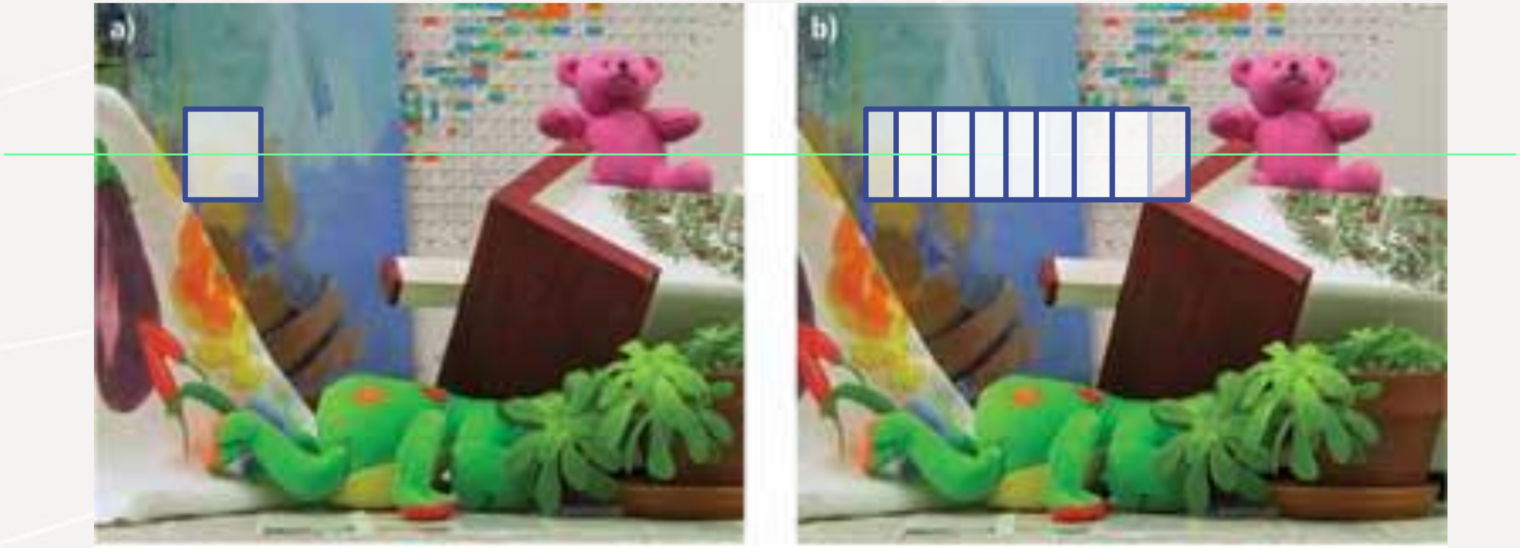
Cost Calculation

- Cost calculation (CC, matching cost) (e.g. absolute difference, squared difference) for each pixel pair in given support window for each disparity level
- E.g. for 7x7 window and SAD, there are 49 absolute difference calculations



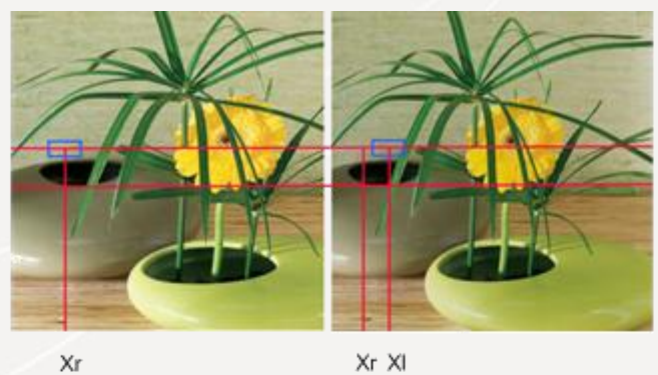
Cost Aggregation

- Cost (support) aggregation, sum of matching costs for a support window at a given disparity level
- For 7x7 window and SAD, a sum of the 49 absolute difference calculations repeated for each disparity level, e.g. for 64 disparity levels



Example: SAD Algorithm

- Calculate sum of absolute difference for each pixel in matching blocks (windows) between left and right image. Repeat for a range of pixels out to a maximum disparity range.
- Determine minimum SAD (shown in graph), for each pixel pair (block).



$$SAD(k) = \sum_{i=0}^{n-1}\sum_{j=0}^{m-1}\left|Right_{IN}(i+k, j) - Left_{IN}(i,j)\right|$$

Disparity: $d = Xl - Xr$

Disparity Mapping Loading

CC: cost calculation

CA: cost aggregation

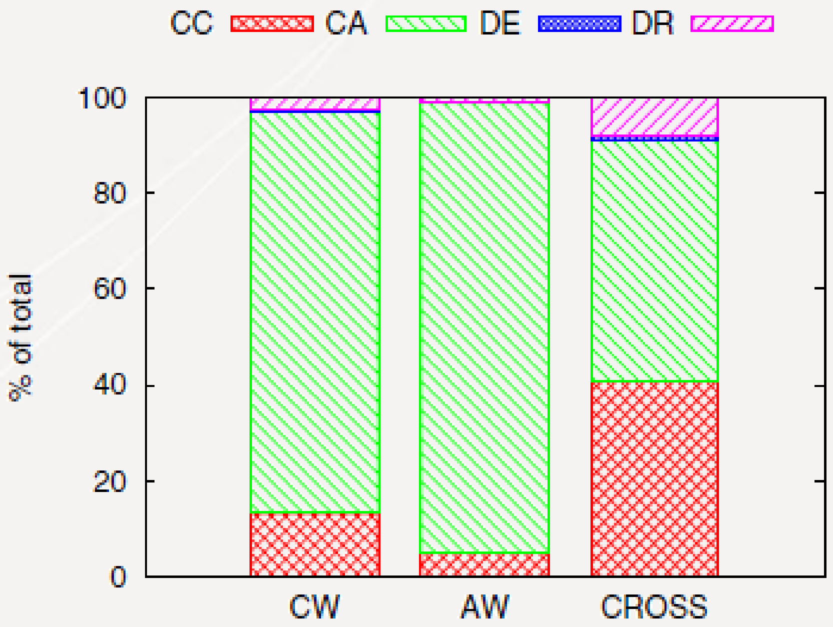
DE: disparity estimation

DR: disparity refinement

CW: constant window

AW: adaptive window

Cross: horizontal and vertical pixel arrays for aggregation



Fang et al, 2012

Magnitude of Performance Required

- Number of cycles per frame for dense disparity is dependent on
  - Pixels per frame,
  - Size of window for window cost calculation,
  - Disparity levels (max disparity assumption)
- Assuming:
  - 1280x960 frame size, 7x7 window, 64 disparity levels
  - Equivalent to ~800 GOPs/sec at 30 fps
- Performance results
  - APEX-1284 achieves 34 fps at ~2700 MDE/s with above assumptions <200mW

Some DE Performance Examples

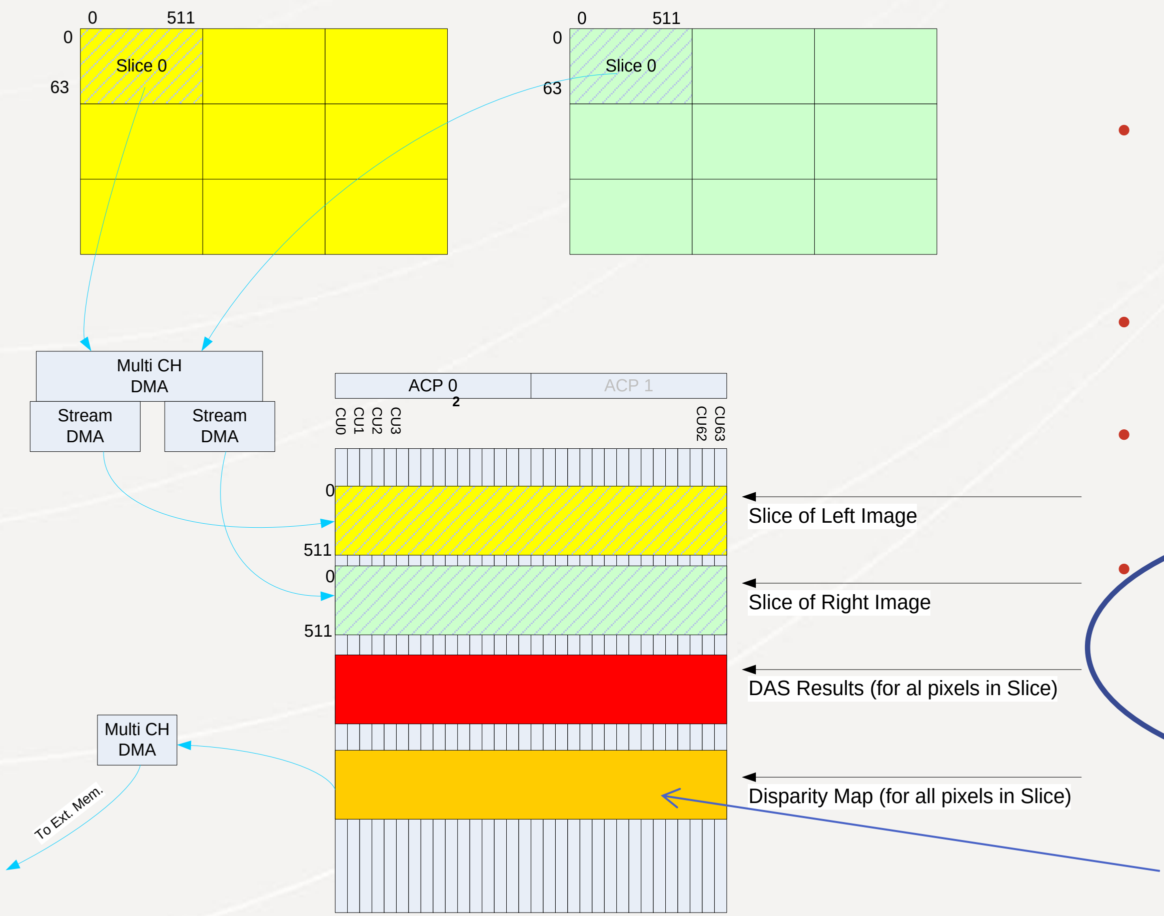
Assuming: 320x240 frame size, 32 disparity levels, 16x16 block size

Method	GPU	Cores	Power	MDE/s	FPS
Local, Constant SW Size	APEX-1284 500MHz*	128	<200mW	995.0	405
Kowalczuk et al 2012	GeForce GTX 580	512	>200W (card power)	152.5	62
FastBilateral	Tesla C2070	448	>200W (TDP)	50.6	21
RealtimeBFV	GeForce 8800 GTX	128	185W (TDP)	114.3	46
RealtimeBP	GeForce 7900 GTX	-	>300W	20.9	8
ESAW	GeForce 8800 GTX	128	185W (TDP)	194.8	79
RealTimeGPU	Radeon XL 1800	-	160W (desktop)	52.8	21
DCBGrid	Quadro FX 5800	240	189W (desktop)	25.1	10

From Kowalczuk et al 2012, see references in source paper

* not part of Kowalczuk et al 2012 results, added for comparison purposes

Architectural Approach with APEX



- DMA transfer slice to CMEM.
  - Interrupt Sequencer when done
  - Stream DMA arranges data runtime.
- ACP, polling Sequencer,
  - Start fetching Slice N+1 (Multi Buffer pipeline)
  - start SAD algorithm (on Slice N)
- Disparity Map generated for Slice N
- ACP trigger DMA to transfer Slice Disparity Map to Ext. Mem.
- As SAD(k)/pixel is generated, compare with previous SAD(k) and check for minima. Store {k, minima} value only.

Algorithm Optimizations

- Simple block matching technique (as described above) is more compute intensive (especially cost aggregation step which is directly related to MNr^2): M is number of disparity levels, N is image size, and r^2 is block size
- Various algorithmic optimizations can reduce computational complexity AND improve quality.
- Chowdhury and Bhuiyan, 2009 used a disparity threshold and "average" disparity" to reduce cost aggregation.
- An integral image approach (Facciolo et al, 2013) reduces computational load to MN (evaluate matching cost at each pixel in constant time)

Using Integral Image

- Difficult to guess the “best” constant window size for local dense disparity mapping
- Varying window size (Adaptive Support Window) increases the quality of local dense disparity mapping approach but window size impacts computational load
- Integral image makes computation load independent of window size
- CogniVue APEX Integral Image benchmarking is significantly better than alternative architectures
- Stay tuned for more disparity mapping updates

Summary

- Multiple approaches for 3D sensing, all generate 3D depth maps. A class of applications (low power, low cost, and sensitive to ambient light) are best supported with stereo image sensor
- Local dense disparity mapping approaches are commonly used in embedded applications
- CogniVue APEX architecture achieves >30fps and ~2700 MDE/sec with megapixel input and dense local disparity mapping with constant support window size at low power
- Further optimizations are available to improve real time performance AND enhance quality of 3D depth map