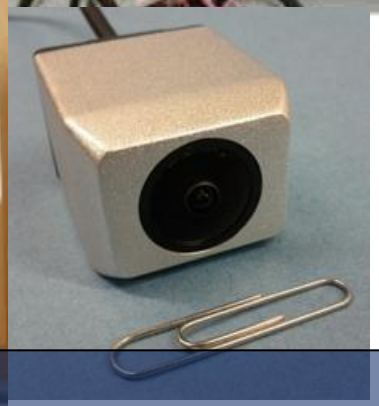
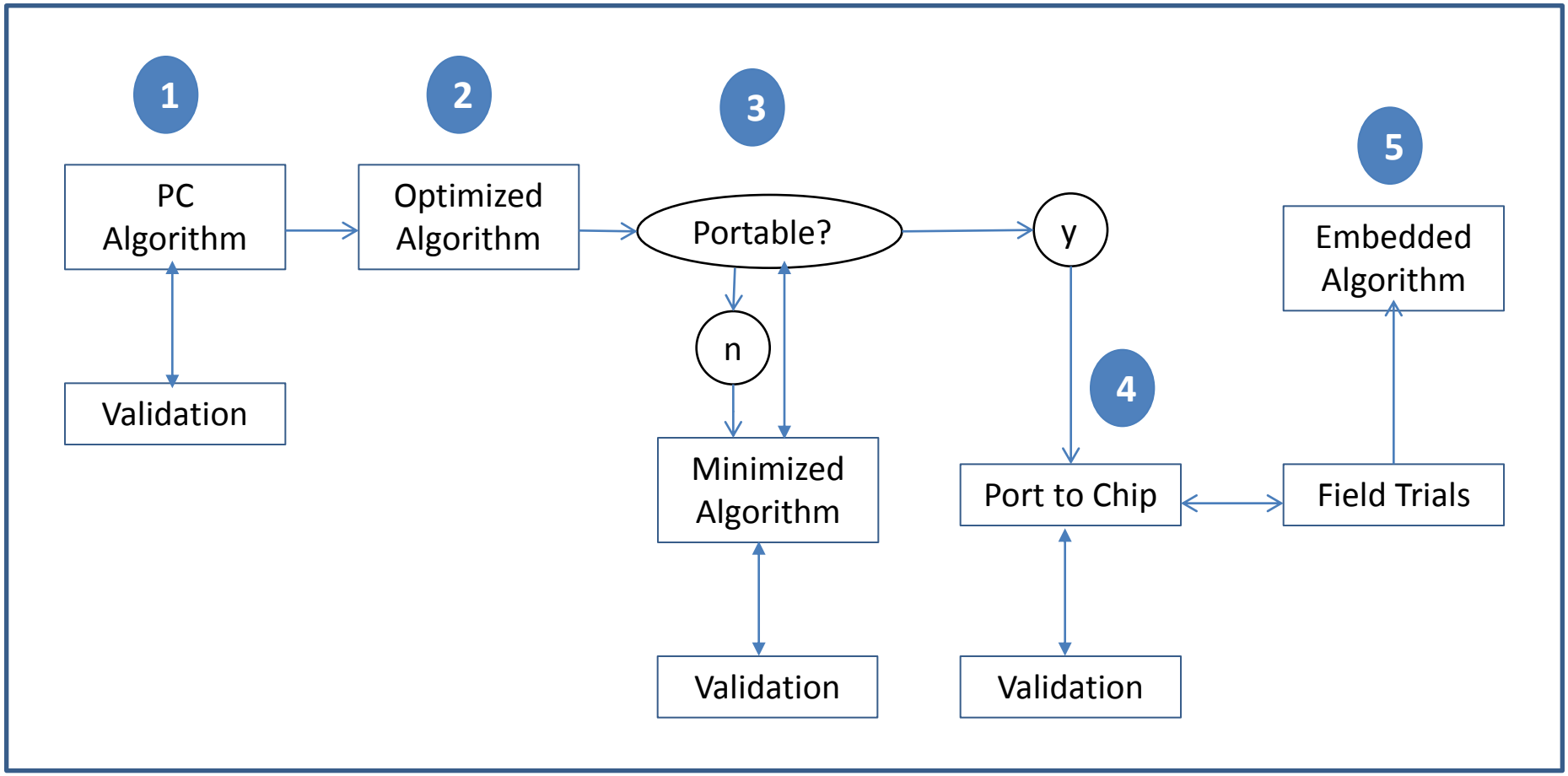
cognivue™
CORPORATION
WE MAKE CAMERAS SMART

embedded VISION SUMMIT

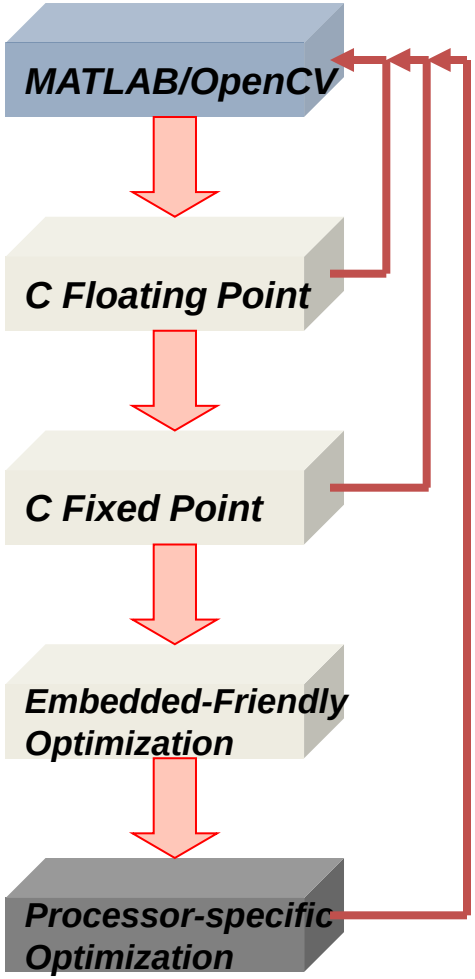


Low-Cost, Low-Power, Small Form Factor Vision Applications: Challenges and Techniques

Embedded Development



Typical Algorithm Optimization



- **C Floating Point**
  - Data dynamic range analysis
- **C Fixed point**
  - Understand & optimize the high level algorithm
  - Data type optimization & error analysis
  - Normalization & scaling
  - Algorithm profiling
- **"Embedded Friendly" Optimization**
  - Inline functions
  - Alignment of data
  - Sequential memory references
  - Optimization of mathematical operations
  - Minimizing memory usage, reliance on large caches
- **Deeply Embedded Platform Specific Optimization**
  - Assembly code (RTL for FPGA)
  - Parallel instructions
  - Pipeline optimization
  - Memory management & data storage
  - Cache optimization

cognivue CORPORATION

embedded VISION SUMMIT

What the Customer Wanted

- Better rear camera, attractive ASP, BoM < $35
- De-warp 180 degree fish-eye, offer multi-view with overlay
- Track the closest object, replicate ultra-sonic zones
- Use a single VGA WDR sensor, 180 degree fish-eye lens
- From boot-up to object detection in <500msec
- Processing in the camera module

Algorithm Development

- Research determined (started on PC with OpenCV):
  - Need for 2 separate obstacle detectors (static, moving)
  - Need for 3D reconstruction of the scene to calculate distance
  - Do distortion correction, over-lay, object detection in real time

- Embedded platform required
  - Partition of image processing primitives – scalar and vector
  - Simplifying data structures to remove unnecessary de-referencing, function calls, and virtual functions
  - Replacing C++ STL objects (*vectors, lists, shared pointers*) used in OpenCV with simpler data structures
  - Accounting for time dependencies of simultaneously running tasks

Validation and Testing

Significant time is needed to build video test data base, 100+ video scenarios

Three levels of testing needed:

1. Algorithm assessed by observing the detection and false detection rate over the entire video test database
2. Testing on the embedded platform
    - Check bit trueness with PC version. Helps to Isolate any problems in the porting.
    - Validate on the test database and compare performance with PC version.
3. Perform field trials

Hardware

Primary Challenge

- Module needed to be small - ~ 1 inch cube, fit typical installations
- Module is water tight seal, no internal/external airflow – chips overheat, vision processor needs to be <500mW
- Heat dissipation critical. Image sensor is sensitive to heat

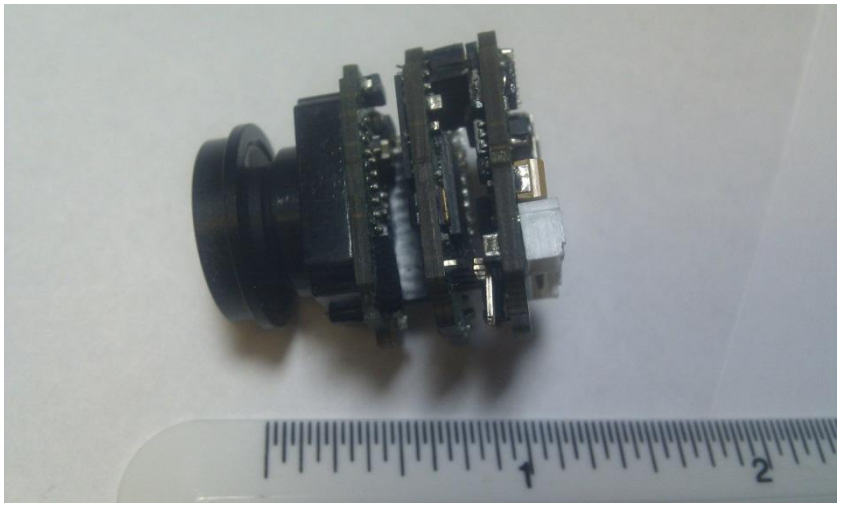
Solutions:

- Several small PCBs. The parts with the most heat kept furthest from sensor
- Metal casing and thermally conductive padding used to contact the case directly to sink heat to the case
- High efficiency DC-DC convertors used (90%+) to minimize heat dissipation in the power supplies. Passive LDO regulators only used with low in-out voltages for image sensor analog supplies

Result: Smart Rear Camera

Within a 1” cube, image cognition processor inside rear camera can do:

- Distortion correction and perspective view
- Feature detection, tracking and distance estimation
- Overlay of lines, text and NTSC/PAL analog video output



Lessons Learned and Applied

- Programming guidelines:
  - Programming guidelines for developers – reduces re-writes
- Vision centric software framework:
  - Algorithm development does not consider data movement
  - Framework ensures data is available for processing
  - Reduces pipeline stalls, cache misses
- Specialized vision libraries:
  - To speed porting on multi-core and vector processors create higher level complex processing “primitives” (specialized vision libraries)

Lessons Learned and Applied

- Tool for camera calibration & LUT generation
  - Task of generating LUTs for sensor/lens combo is tedious and time consuming
- Need for comprehensive golden image test database
  - Create in parallel with algorithm development, automate testing
  - Determines if algorithm changes have impacted performance
  - Validate the algorithm against test data base prior to porting
  - Variability in the in-field environmental conditions will alter algorithm performance. Field trials must use the same test criteria
- Plan to do several iterations

Thank-you, Merci,

Danke Schon, Tack Sa Mycket,

ありがとう,

Kam Sa Hum Ni Da, 감사합니다,
Duo Xie, 谢谢

www.cognivue.com

+1 819-595-4999 x200