Efficient Super-Resolution Algorithms and Implementation for Constrained Applications

Ilan Yona, Imaging & Computer Vision Director

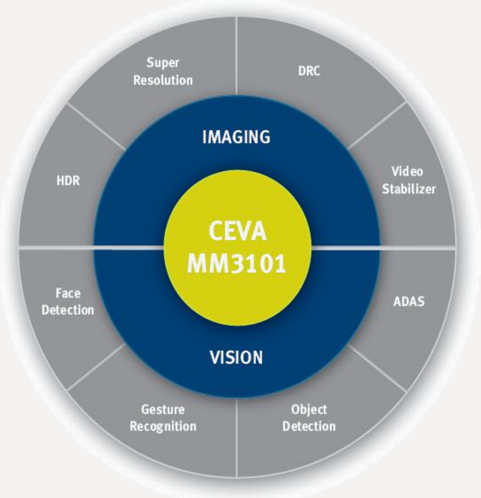
October 2, 2013

Outline

- Company Introduction
- Scope and Motivation
- The Embedded Challenge
- Non-uniform Interpolation Approach
- CEVA's Solution
- Quality Comparison
- Porting Algorithm to a Real-Time Embedded Solution
- Performance Comparison
- Summary and Conclusions

Company Introduction

- CEVA—The world's leading licensor of DSP cores & platform solutions for mobile, home and automotive markets
- 4.5 billion CEVA-powered products have shipped to date
- CEVA-MM3101 is a fully programmable Imaging and Computer Vision DSP IP Platform
  - 3rd Generation multimedia DSP IP Product
  - Application Dev-Kit including CV libraries, abstraction of CPU offloading
  - Algorithmic products including SR, DVS
- Embedded-Vision Alliance founding member



Scope and Motivation

- Super-Resolution (SR) Challenge - Create a high resolution image using several low resolution images
- Market Requirements
  - Create high quality and high-res pictures from low resolution sensors
  - Enable quality zoom
  - Eliminate noise

Low res sensor

SR Algorithm

High quality image

Scope and Motivation

Can CV applications Benefit from SR output?

The Embedded Challenge

- Embedded systems are restricted in terms of:
  - Processor frequency
  - Power consumption
  - Memory on chip
  - Memory bandwidth (BW)
- The chosen algorithm for any application must carefully consider the above restrictions in order to be able to function on portable devices in real-time environment

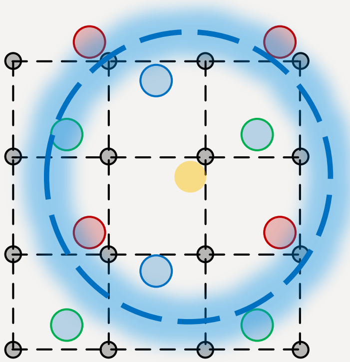
How Can We Increase Image Resolution?

- Popular methods for multi-image Super-Resolution:
  - Iterative Back-Projection Based
  - Frequency Domain Based
  - Normalized Convolution
  - Statistical methods
  - Non-Uniform Interpolation

CEVA

Non-Uniform Interpolation Method

- Each high-res pixel has a group of low-res neighbors
- They are distributed non-uniformly
- Find set of coefficients (weights) for linear interpolation



- Performance/Bandwidth advantages:
  - Easy to parallelize calculations (no dependencies)
  - Simple filter (using MAC)—Efficient in DSP cores
  - Serial data access—Data can be accessed in continuous "raster like" order
  - Non-iterative—Bounded and predicted calculation time

SR Non-Uniform Interpolation

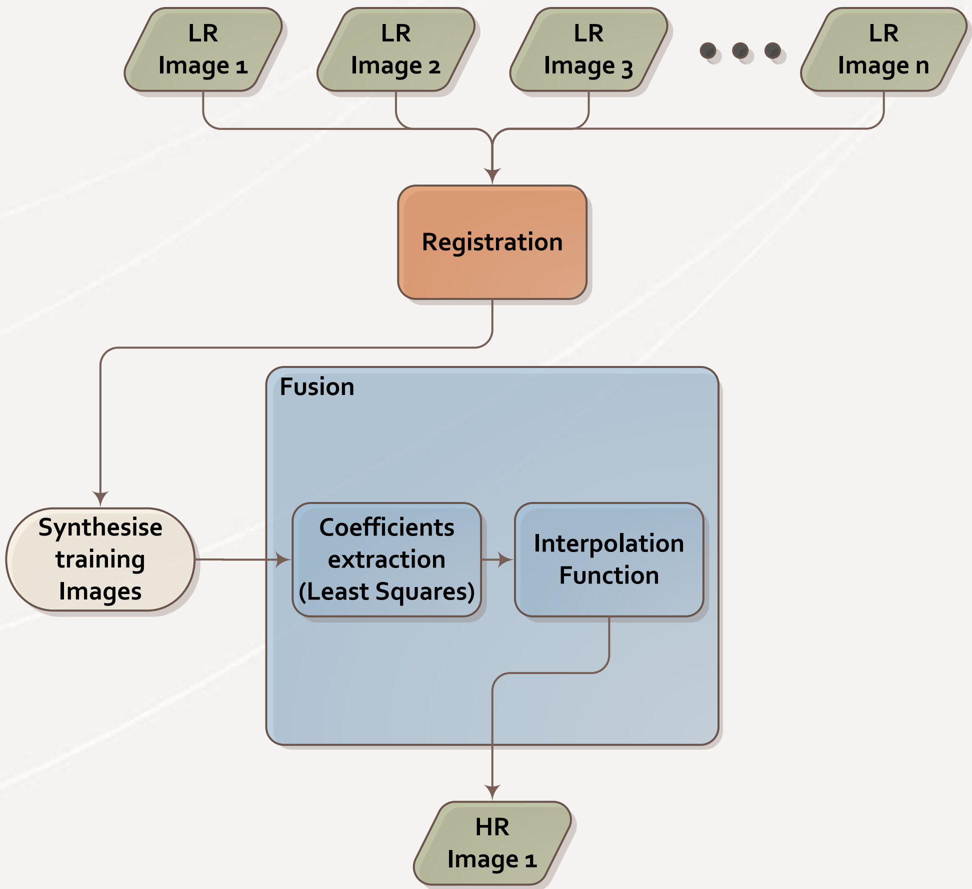
A. Gilman, and D. G. Bailey (GB), “Near optimal non-uniform interpolation for image super-resolution from multiple images”

- What is the big news?
  - The required interpolation weights are quite independent of the image content
  - The weights depend on the relative position of the input images
  - One can train an interpolation kernel (from known images) and apply it on different images and scenes
- Cons:
  - Training is an expensive process

    => doesn’t fit embedded devices

CEVA

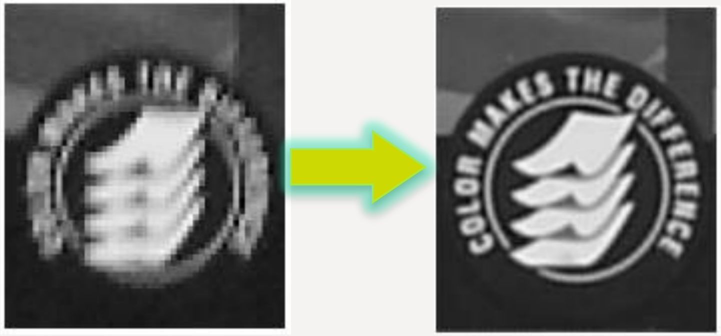
Non-Uniform Interpolation—GB System Overview



SR Non-Uniform Interpolation (2)

Analytic method (Michaeli & Eldar SSP'09)

- Derive optimal filter based on priors (eliminates need for training)
- Assume prior knowledge on
  - Scene spectrum (ACF)
  - Camera system (PSF)
  - Sensor noise statistics
  - LR image displacement
- Cons
  - Assume infinite support
  - (continuous signal)

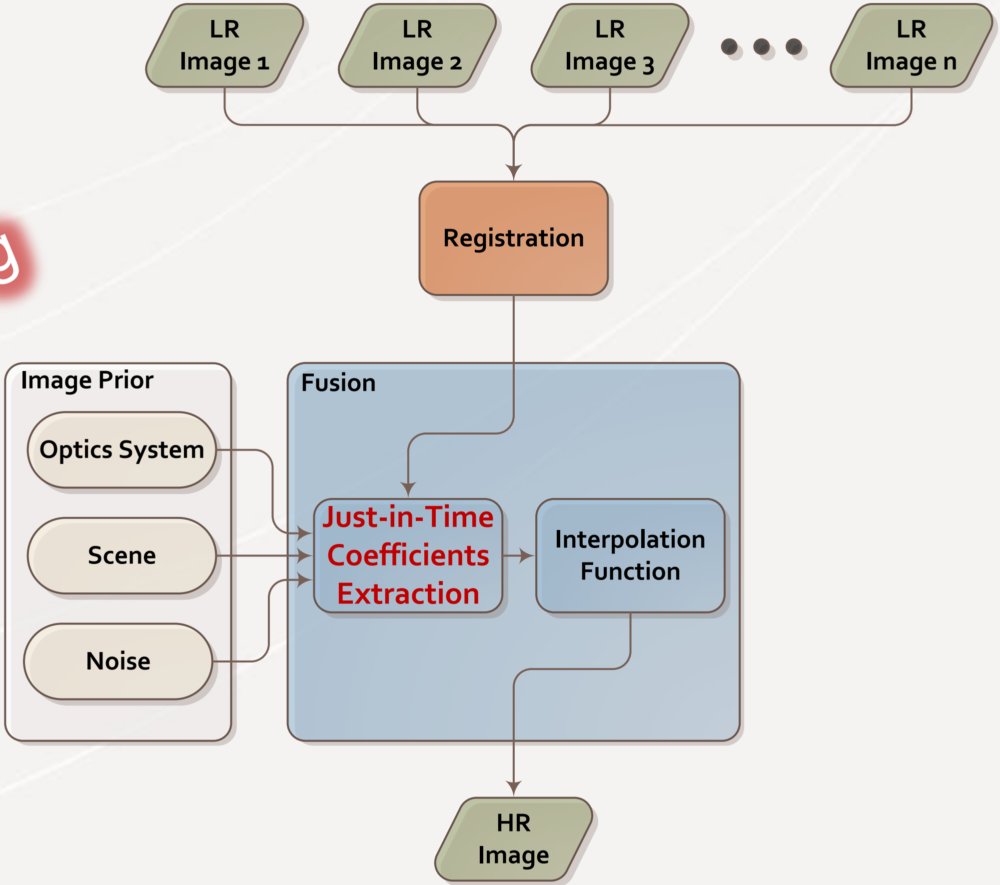


CEVA's SR—Combined Solution

Empirical Solution Gilman & Bailey	Analytic Solution Michaeli & Eldar
✗ Training	✓ Analytic and Robust
✓ Finite Support	✗ Infinite Support

Combined Solution CEVA SR
✓ Analytic and Robust
✓ Finite Support

CEVA's SR—System Overview



No Training

CEVA's SR Concept

1. Divide the image into small tiles
   - Enables each tile to have different displacement, PSF, ACF and noise
   - Enables efficient local memory utilization

2. JIT coefficients extraction

   Coeff(i,j) = CEVA_FAST_COEFF (Displacement, PSF, ACF, Noise)
   - Build model equations
   - Solve them JIT

CEVA's SR Concept (2)

3. Main JIT processing contributor is matrix inversion
   - Matrix inversion module can be challenging for standard vector processors
     - Large matrix handling
     - Nested loops with variable iteration counters
     - Parallel sparse memory accesses

=> MM3101 architecture overcomes these issues and efficiently utilizes the vector processing units

Quality Comparisons

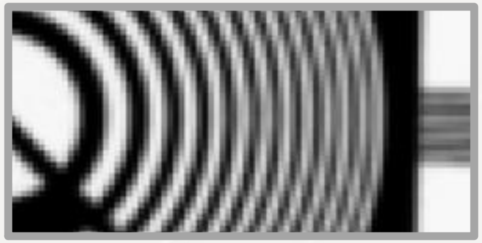
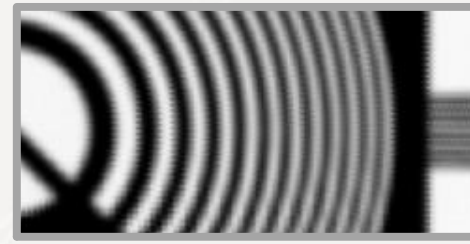
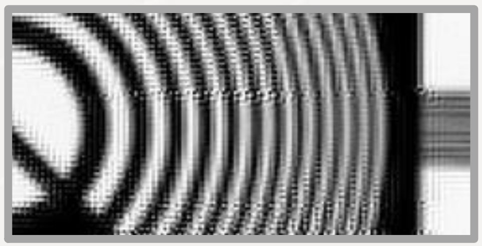
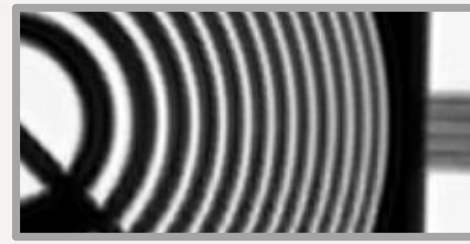
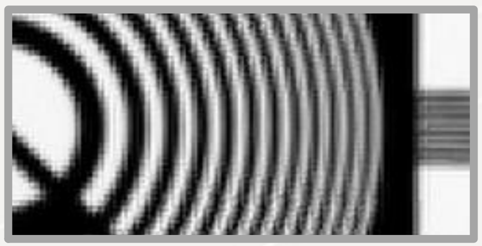
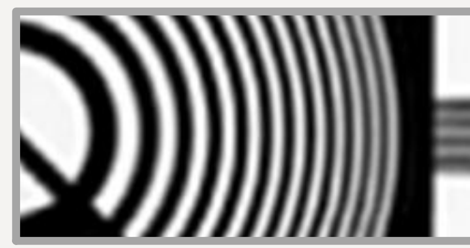
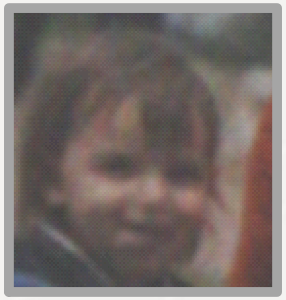
Bi-Cubic		**Structure Adaptive Normalized Convolution**	
Iterated Back – Projection		**Commercial PC App**	
Robust Super Resolution		**CEVA**	

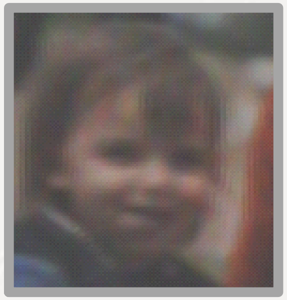
Image from ISO 12233 test pattern—LCAV - Audiovisual Communications Laboratory.

Quality Comparisons (Cont.)

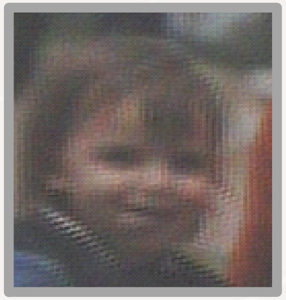
Bi-Cubic



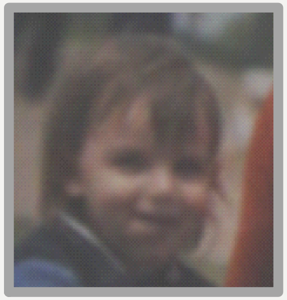
Structure Adaptive Normalized Convolution



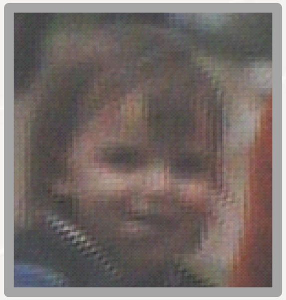
Iterated Back Projection



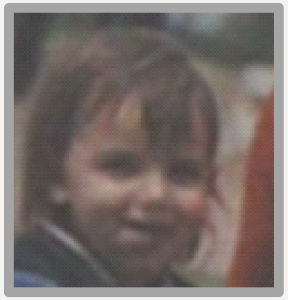
Commercial PC App



Robust Super Resolution



CEVA



Photograph taken with a Canon 550D camera [LCAV—Audiovisual Communications Laboratory.](#)

Super-Resolution & Computer Vision

- Can SR implementation benefit from CV kernels?
  - Yes!
  - Accurate registration is a very important phase
  - The relative position precision requires sub-pixel accuracy (e.g. 1/16)
  - Features extraction and tracking → Accurate/Cost effective
    - Sobel filter
    - Harris corner detection
    - GoodFeaturesToTrack
    - KLT

Porting Algorithms to a Real-Time Embedded Solution

- CEVA's MATLAB algorithms are using popular computer vision and image processing kernels such as
  - Common Interpolations:
    - Bi-cubic
    - Bilinear
  - Optical Flow:
    - Sobel filter
    - Harris Corner Detection
    - KLT
  - Filters and Convolutions:
    - Cross Correlation
    - FIR filtering
- All of the above are part of the CEVA-CV library (OpenCV-like libraries)
  - Immediate and optimized porting for large portions of the algorithms

SW Development Acceleration - SmartFrame

- CEVA offers a SW module (SmartFrame) that enables using imaging/CV kernels with minimal awareness to the system architecture
- This module handles:
  - External/internal memory allocation
  - Number of in/out buffers in the TCM
  - Tunneling of kernels
  - Surrounding pixels required for each kernel
- According to the above, it activates automatically the whole system
  - Allocating memory in the TCM
  - DMA operations
  - Running the kernels (including tunneling in tile level)

SR port to CEVA-MM3101 platform completed in under 2 months!

Performance Comparison

Method	*Complexity [op / pixel]*
Iterated Back-Projection	**More than 10,000**
Robust Super-Resolution	**More than 12,000**
Structure Adaptive Normalized Convolution	**Very complex: SVD for every pixel**
Commercial PC Application	**100-400**
CEVA SR on CEVA-MM3101	**<100**

Matching Real-Time Embedded Solutions

- Ported SR algorithm to CEVA-MM3101 platform
- Performance includes registration and fusion phases

  Use case: Input: four 4MPixel
  Output: 16MPixel 4:2:2

- Power: as low as 20mW (@28nm)
- DDR bandwidth: <80MB
- Real-time processing: Can output multiple images in a second

Summary and Conclusions

- When dealing with CV in restricted systems, a high quality algorithm is not enough
- In general, many good algorithms can be found for each **challenge** or problem

But what is the best one for **you?**

Summary and Conclusions (2)

- Verify carefully what are the market use cases/requirements
  - Power consumption limit
  - DDR BW limitation
  - Cycle count limitation
- Take into considerations your processor strengths and weaknesses
  - TCM size
  - Special instructions
  - DDR access time
  - Cache size
- TTM—Choose algorithms that enable re-use of existing optimized modules in the system

Summary and Conclusions (3)

- The algorithm that was chosen enabled us to:
  - Process the images tile by tile
    - Effective for good DDR utilization
    - Data transfer as background task
  - Non-iterative process
    - All pixels in a tile handled in same manner, in single iteration
    - Utilize vector engine effectively, handles multiple pixels simultaneously
- Short TTM by utilizing:
  - Existing CEVA-CV (OpenCV-like libraries)
  - SmartFrame module

Software-Based Super-Resolution Technology for Low Energy Embedded Applications

Resources for Further Investigation

- Academic references:
  - Iterated Back-Projection. M. Irani and S. Peleg, Graphical Models and Image Processing, 1991.
  - Robust Super-Resolution. A. Zomet, A. Rav-Acha, and S. Peleg, CVPR, 2001.
  - Normalized Convolution. Tuan Q. Pham, Lucas J. van Vliet and Klamer Schutte, EURASIP Journal on Applied Signal Processing, 2006.
  - T. Michaeli and Y. C. Eldar, "The vector hybrid Wiener filter: Application to super-resolution," *IEEE Workshop on Statistical Signal Processing* (SSP 2009).
  - A. Gilman, and D. G. Bailey, "Near optimal non-uniform interpolation for image super-resolution from multiple images," 2006
- LCAV - Audiovisual Communications Laboratory, http://lcav.epfl.ch/software/superresolution
- Embedded Vision Alliance resources—http://www.embedded-vision.com/

WWW.CEVA-DSP.com

Thank you

are all invited to the CEVA demo table