JPEG 2000 Part 2 Progress Report: Enabling Support for Arbitrary Filter Banks

Chris Brislawn and Brendt Wohlberg

Los Alamos National Laboratory

July 2000

WG1 document number WG1N1820

LANL document number LAUR-00-3317

Objective

- Allow user-supplied filter banks other than the 9/7 and 5/3 from Part 1
- Facilitates various extensions:
  - Specific camera or computing hardware characteristics
  - Specialized image content
  - Multicomponent band-decorrelation filters
  - Data-dependent filters (e.g., “principal component filter banks”)
- Decision to employ user-supplied filter bank is an *encoder* option; hard to anticipate in advance what encoders will want to do in the future, but syntax and algorithms for decoding need to be supplied in Part 2.
- Two primary categories of “Part 2” filters:
  - “Old favorites” that have been used in the past
    - Can codify specific implementations, including reversible methods
  - Arbitrary user-defined filter banks
    - Need generic algorithms for implementing and signaling filter banks
    - Demand for both odd- and even-length linear phase filter categories

Key Technical Problems

- Implementation of generic odd-length filters is a very straightforward generalization of current Part 1 algorithms:
  - Same symmetric extension methods used for the 9/7 filter bank will work for arbitrary odd-length filter banks, either convolutional or lifted.
  - Symmetric extension does not affect reversibility; i.e., symmetric extension techniques enable lossless coding with *any* reversible filter bank.
  - Lifting implementation probably necessary for reversible transforms.
- Signaling generic lifting implementations requires validating a predefined, generic lifting architecture:
  - VM currently accepts high-order lifting steps
  - May be problem with current factoring conventions for certain filter lengths
  - Has signaling of reversible filter banks been tested in the VM?
- Implementation of generic even-length filters requires a new algorithm to enable resolution scalability.

Resolution Scalability for Even-Length Filter Banks

- Resolution scalability: expressed by the convention of alternating lowpass & highpass samples, even across tile boundaries, regardless of tile sizes or number of levels of wavelet decomposition.
  - Equivalent to having DWT algorithms for both "lowpass-first" and "highpass-first" transforms on both even-length and odd-length tiles:
    - For even-length tiles (N=2K), need algorithm that produces K lowpass and K highpass outputs. (For even-length filters, just one such algorithm suffices for both the lowpass-first and highpass-first cases.)
    - For odd-length tiles (N=2K+1), need 2 algorithms, one that produces K+1 lowpass and K highpass outputs ("lowpass-first"), and another that produces K+1 highpass and K lowpass outputs ("highpass-first")
  - This is already enabled for arbitrary odd-length filter banks using symmetric extension algorithms from Part 1.
- With even-length filters, *symmetric* extension doesn't allow production of K+1 highpass and K lowpass outputs from odd-length inputs.
  - Consequence of having an antisymmetric highpass filter
- Solution: define highpass-first transform for odd-length inputs by using (2,2)-*antisymmetric* extension of the input.

Results to Date

- LANL has implemented the symmetric and antisymmetric extension algorithms for even-length linear phase filter banks in VM 7.2.
- Allows decomposition using even-length filter banks on arbitrary-size images with arbitrary tilings, subband decompositions, or reference grid offsets.
  - Appears to handle all cases successfully based on testing done to date.
- Implemented via convolutional filtering modules in VM 7.2.
  - Method not constrained to convolutional transforms; equally applicable for lifting implementations and reversible even-length filter banks.
- We have not experimented with image manipulation or trans-coding algorithms that utilize resolution-scalable DWT representation on the reference grid.
- Software will be posted to the WG1 web site after the Arles meeting.
- We would like to add this capability to the VM and Part 2 WD. A core experiment on this topic, CodEff03, has been registered for Rochester.

Conclusions

- A number of established filter banks, both odd- and even-length, have been used in JPEG experiments and elsewhere; many have found niche applications. They should be made available to users via Part 2 options.
- Signaling generic lifting implementations may be feasible (albeit possibly with some limits on complete generality) and would allow signaling of reversible implementations, but work remains to be done to establish sufficiently general methods.
- Possible to enable resolution scalability for both odd- and even-length filter banks solely via {anti}symmetric extension methods.