Fast Intra-Prediction Mode Selection for H.264

Jason Fritts, Huakai Zhang, Hui Zhang, Frederick Steiling

Presented by Hui Zhang
Introduction
H.264

- The next generation in video coding and compression technologies
  - Advantages
    - Enhanced visual quality at very low bit rate
    - Better compression through advanced compression techniques
      - Intra-prediction
      - Rate-Distortion optimization
      - Context-adaptive entropy coding
      - Built-in deblocking filter
  - H.264 coding/decoding is complex and takes time
    - Requires speedup through algorithm optimization
Block Diagram (H.264 coder)
Intra-Prediction
Intra-Prediction

- **Motivation**: intra-coded frames in natural images exhibit strong spatial correlation

- Blocks/macroblocks in intra-coded frames can be predicted from previously-coded blocks/macroblocks
  - Above and/or to the left of the current block

- An encoded parameter specifies which neighbors should be used for prediction, and how
Intra-Prediction

- Intra-coded blocks/macroblocks can now be predicted from neighboring blocks/macroblocks
- Only the difference between the predicted and coded block/macroblock needs to be encoded
  - 4 luminance prediction modes for 16x16 macroblocks
  - 9 luminance prediction modes for 4x4 blocks
  - 4 chrominace prediction modes
4x4 Block Luminance Intra-Prediction Modes

- 8 directional prediction modes
  - block M is predicted from 1 or more of its four previously coded neighboring blocks (A, B, C, D)

- 1 DC prediction mode
  - only mode available in prior standards
4 x 4 Intra Prediction (con’t)

- 4x4 prediction pixel values are interpolated from the 13 neighboring pixels (A-L and Q) according to the prediction direction.
Example of Pixel Interpolation for Intra-Prediction Modes
Fast Intra-Prediction Mode Selection
Heuristic: Correlation-Based Prediction Table

- Basic Idea:
  - Reduce number of 4x4 intra-prediction modes to check
    - i.e. Instead of checking all 9, only check 1, 2, or 3…
  - 4x4 intra-prediction modes are correlated, so use intra-prediction modes of neighbors to estimate intra-prediction mode of current block
  - Create a correlation table from a set of training video sequences, which gives the best prediction modes for each specific set of neighboring nodes
Correlation Table

- Correlation table:
  - Correlate prediction off of left and above neighbors
    - can correlate prediction off of all 4 previously-coded neighbors (left, above, above-left, above-right), but this requires a very large table
  - Save ordered set of prediction modes, from best to worst, for each specific left/above neighbor pair
  - When looking up the prediction modes for a given 4x4 block, if one of the neighbors is unavailable, use mode 2 (DC mode) as that neighbor’s mode

\[
\text{correl } [\text{left\_mode}] [\text{up\_mode}] = \{4, 1, 8, 6, 3, 0, 2, 7, 5\}
\]

\[\uparrow \quad \uparrow\]

best mode       worst mode
System Architecture

Training Inputs → H.264 Encoder → Compressed Output

Correlation Statistics

Generate the correlation table

Apply the correlation table

Evaluation Inputs → H.264 Encoder → Compressed Output
Proposed Heuristic

Original version:

```c
for ipred_mode in 0 to 8 {
    find_best (ipred_mode);
}
```

```c
find_best_mode (int ipred_mode) {
    if (mode_cost = diff_cost (ipred_mode) < min_cost) {
        best_mode = intra_pred_mode;
        min_cost = mode_cost;
    }
}
```

New heuristic:

```c
for i in 0 to NUM_MODES_TO_CHECK {
    ipred_mode =
        correl[left_mode][up_mode][i];
    find_best (ipred_mode);
}
```
Experiment Results
Evaluation Environment

- Used H.264 software verification model JM9.0
  - encoding parameters based on default configuration included with JM9.0

- Ran experiments on a 2.2 GHz mobile Pentium IV
  - timing results measured using `gprof`

- Three versions of heuristic:
  - `pred1` – checking the 1st predicted best mode (from correlation table)
  - `pred2` – checking the first 2 predicted best modes (from correlation table)
  - `pred3` – checking the first 3 predicted best modes (from correlation table)

- Performed experiments both with and without rate-distortion optimization

- Video sequences encoded with 4 different encoding formats
  - II – all I frames
  - IPI – repeating IP sequence
  - IBPBI – repeating IBPB sequence
  - IBBPBBPBBI – repeating IBBPBBPBB sequence
Video Sequences

- Used movie trailers for training and evaluation test sets
- Timing measurements used only 9-10 frames

<table>
<thead>
<tr>
<th>Evaluation Set</th>
<th>Resolution</th>
<th>Frames Used</th>
</tr>
</thead>
<tbody>
<tr>
<td>Friday Night Lights</td>
<td>640x336</td>
<td>916 to 1115</td>
</tr>
<tr>
<td>Lemony Snicket</td>
<td>640x352</td>
<td>315 to 514</td>
</tr>
<tr>
<td>Ray</td>
<td>720x400</td>
<td>915 to 1114</td>
</tr>
<tr>
<td>Terminator 3</td>
<td>640x352 (orig. 640x360)</td>
<td>150 to 249 and 370 to 469</td>
</tr>
<tr>
<td>Wimbledon</td>
<td>720x400</td>
<td>1384 to 1583</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Training Set</th>
<th>Resolution</th>
<th>Frames Used</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bridget Jones 2</td>
<td>640x336</td>
<td>3010 to 3209</td>
</tr>
<tr>
<td>Closer</td>
<td>640x320 (orig. 640x332)</td>
<td>960 to 1159</td>
</tr>
<tr>
<td>Elektra</td>
<td>640x336</td>
<td>916 to 1115</td>
</tr>
<tr>
<td>Hitch</td>
<td>640x320 (orig. 640x332)</td>
<td>2080 to 2279</td>
</tr>
<tr>
<td>Spanglish</td>
<td>640x320 (orig. 640x332)</td>
<td>400 to 599</td>
</tr>
</tbody>
</table>
Best Mode Statistics

- The first three prediction modes are the set of predictions modes most commonly used.
- Collectively, they cover 85-95% of the best modes.
Function Speedup
function: Mode_Decision_for_4x4IntraBlocks

- Speedup of function with RD-optimization are expected to be greater, since RD-optimization is very expensive computationally
Overall Speedup

Overall speedup (with RD-optimization)

Overall speedup (no RD-optimization)
Increase in Bitrate

- Significant reduction in bitrate increase when checking 3 prediction modes
- Increase in bitrate with RD-optimization is nearly identical
Conclusions

- Speedup intra-prediction mode selection kernel by
  - 1.7x to 3.8x when using rate-distortion optimization
  - 3.2x to 7.2x without rate-distortion optimization

- Speedup overall encoding time by
  - 1.22x to 2.66x when using rate-distortion optimization
  - .79x to 1.29x without rate-distortion optimization

- At minimal cost to SNR and bitrate
  - Increase in bitrate is 5-13%
  - Typical change in SNR is < 0.5%

- This method is orthogonal to previous methods
  - i.e. can be used in conjunction with other methods