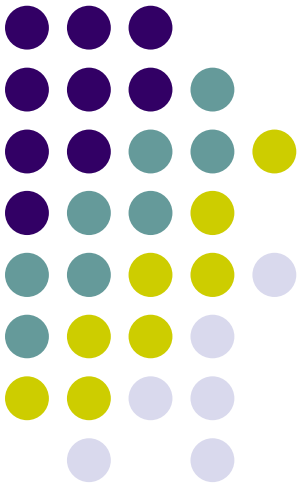
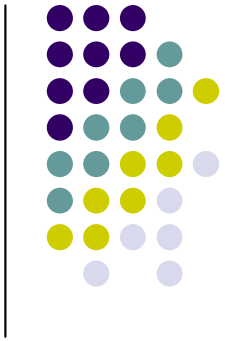
MediaBench II Video:
Expediting the next generation of video systems research

Jason Fritts, Frederick Steiling, Joseph Tucek

Presented by Hui Zhang



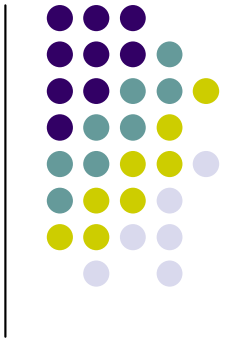
Introduction



- Video is pervasive in the computing industry

- In future video systems, we can expect:
  1) users will continue to demand higher quality video at lower prices
  2) research in information theory will continue to push the envelope in maximizing video compression
  3) designers of video processors and video systems will continue to face the challenge of providing the greatest video capabilities for the lowest dollar

Introduction

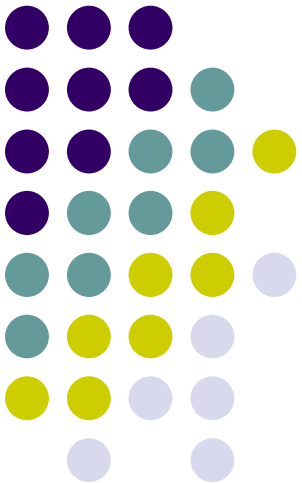


- Design of video processors and systems requires knowledge of:
  - the major video applications and standards, and
  - the workload characteristics of those applications

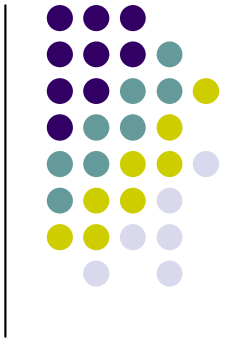
- This knowledge enables systems design decisions, such as:
  1) whether the system will support one or more video standards, and consequently requires hardware and/or software for separate standards,
  2) whether the design should employ more dedicated hardware (which is less flexibile) or more software (which requires more powerful and costly processors), and
  3) what video processing and system control software requirements must be served by the system processor(s)

- MediaBench enables understanding of workload characteristics of video

MediaBench
Now and Then

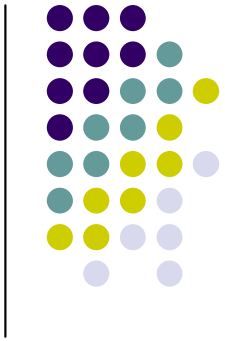


MediaBench



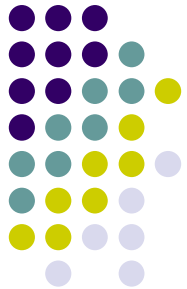
- Introduced in 1997 by Lee, Potkonjak, and Mangione-Smith
- **Goal:** full applications representative of the workload of emerging multimedia and communications systems for computer architecture and compiler research
- Focus Areas
  - video, image, audio, speech, security, graphics
- Applications
  - MPEG-2, JPEG, Epic, Ghostscript, Mesa, ADPCM, GSM, G.721, Rasta, PGP, Pegwit

MediaBench II



- MediaBench: aged, no longer the representative of *emerging* workloads
- Next generation of MediaBench: MediaBench II
- Expanded to distinct application areas:
  - composite + area-specific benchmark suites
- More recent standards added:
  - MPEG-4, H.264, Motion-JPEG2000, etc.

Media-Centric Benchmark Organization



***MB*$_{video}$**
- MPEG-4
- H.26L
- MPEG-2
- H.263

***MB*$_{image}$**
- JPEG-2000
- JPEG
- JBIG?
- TIFF/GIF?

***MB*$_{audio}$**
- MP3
- MPEG-2/4 AAC
- Dolby AC-3
- Dolby DTS

***MB*$_{speech}$**
- GSM
- G.7xx
- Voice recognition

***MB*$_{security}$**
- PGP
- AES

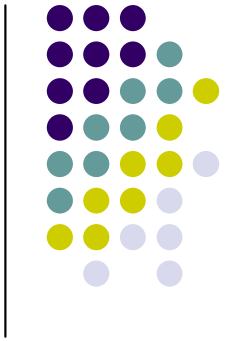
***MB*$_{graphics}$**
- Rendering
- Lighting
- Perspective generation

***MB*$_{analysis}$**
- Segmentation
- Feature extraction
- Character recognition

***MB*$_{kernel}$**
- FIR/IIR
- DCT/FFT
- 2D conv
- Optical flow

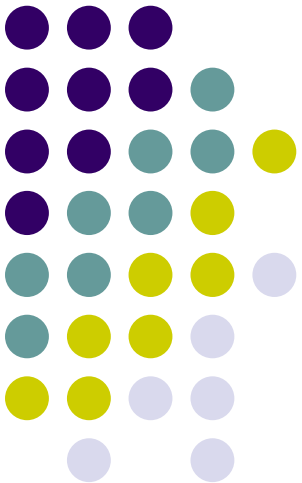
***MB*$_{composite}$** contains 1-2 key applications from each area-benchmark

MediaBench II: Video

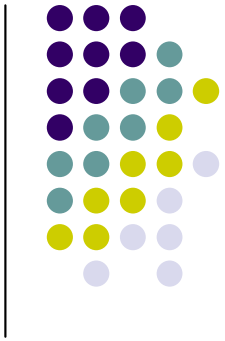


- ***MBvideo*** includes:
  - Popular mainstream video compression standards
    - Motion-JPEG, H. 263, MPEG-2
  - Next-generation standards
    - MPEG-4, Motion-JPEG2000, H. 264

Workload Evaluation
of MediaBench Video



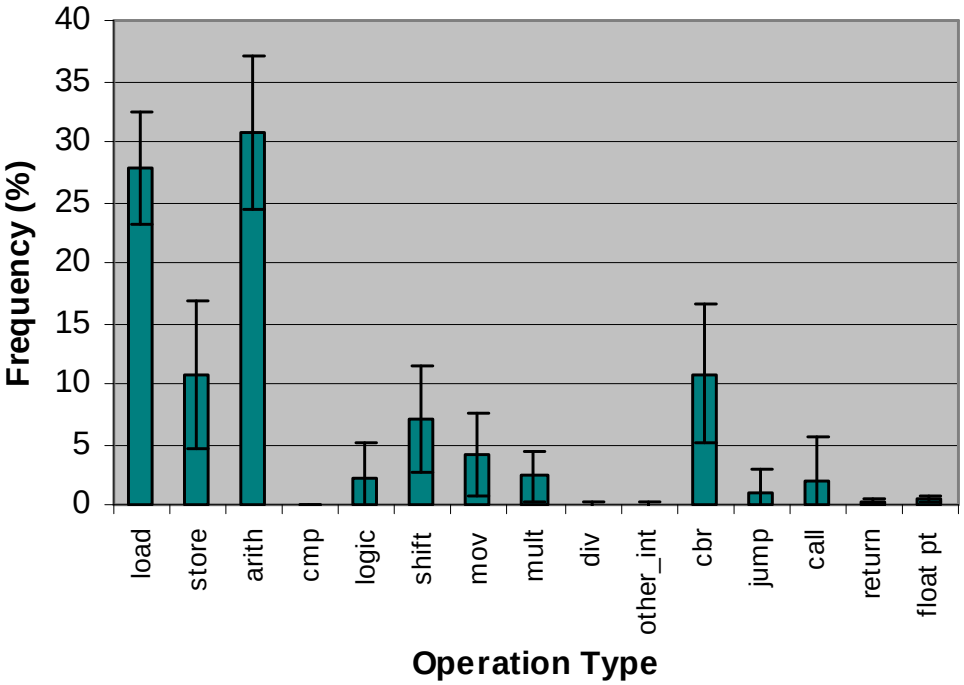
Evaluation Methodology



- Let ***MB**video* speak for itself
  - Quantitatively identify characteristics of MB video applications

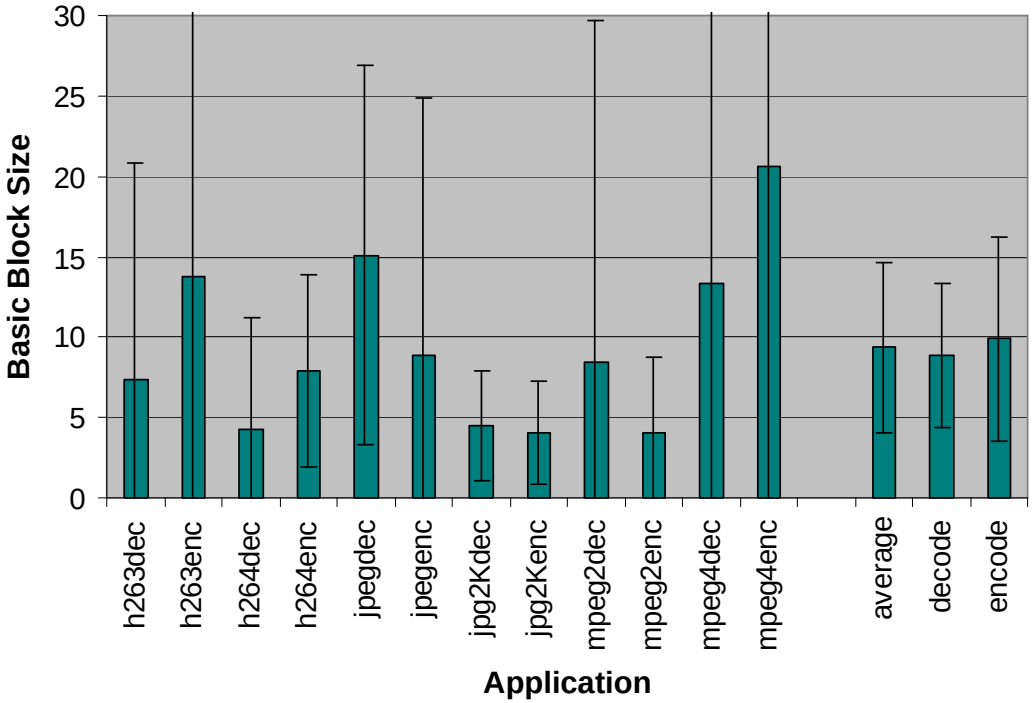
- Evaluation Methodology
  - IMPACT compilation and simulation environment
  - Compile for architecture-independent evaluation
    - targeting a generic RISC ISA
    - using only local compiler optimizations
  - Base video input data set:
    - 4CIF video resolution (706x576)
    - moderate motion
    - medium/average video quality
    - +/- 16 pixels for motion estimation

Instruction Frequencies



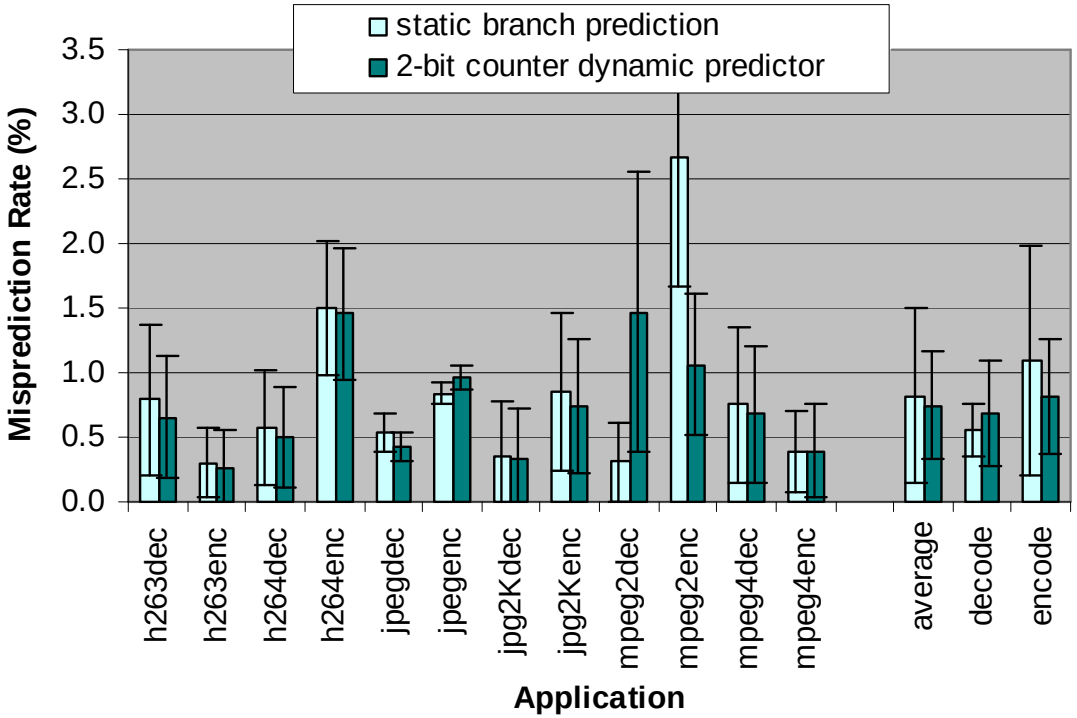
- Vast majority are memory, ALU or control flow instructions
- Major differences with general-purpose applications:
  - minimal use of floating point
  - multiply instruction used only 2.5% of the time
  - lower ratio of control flow operations and compares

Basic Block Sizes



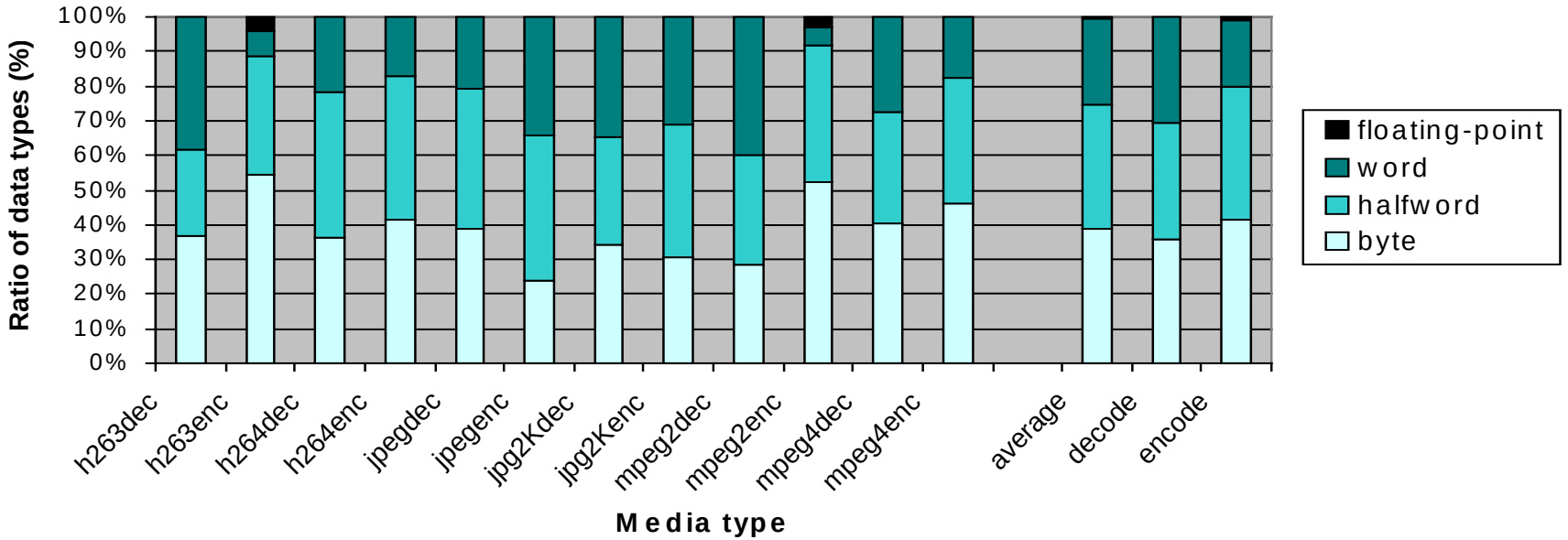
- Representative of maximum amount of ILP
- 9.4 instruction per basic block vs. 5-6 instruction per basic block
- Much greater potential for high ILP

Branch Prediction



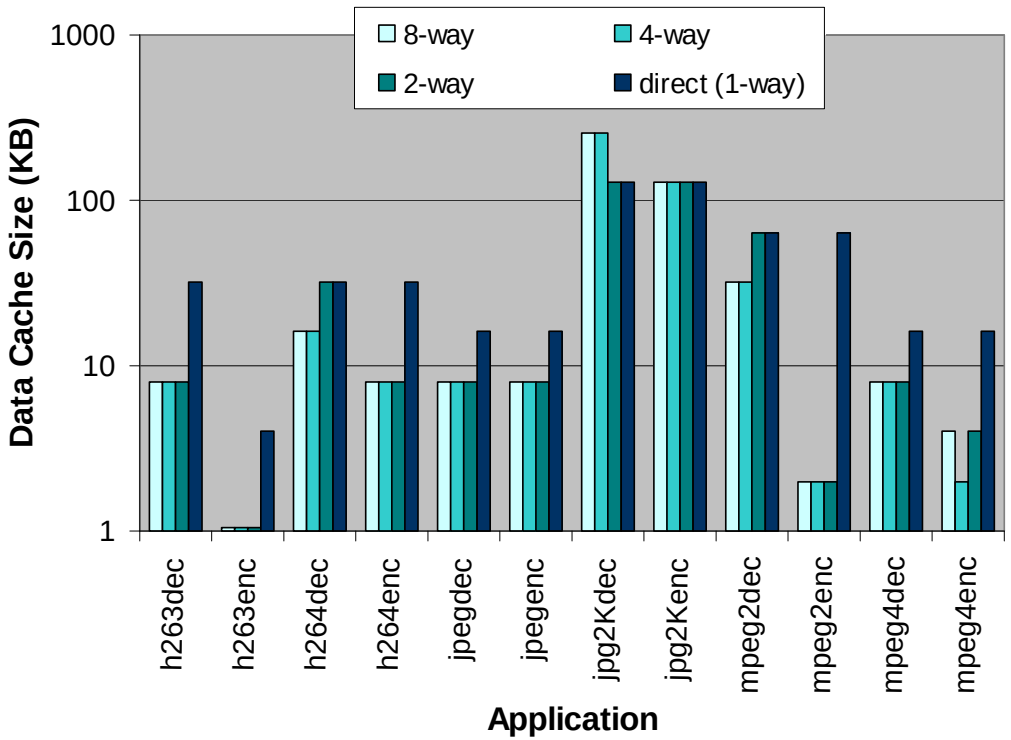
- Exceptional static branch prediction performance
- Indicates significant processing regularity

Data Types and Sizes



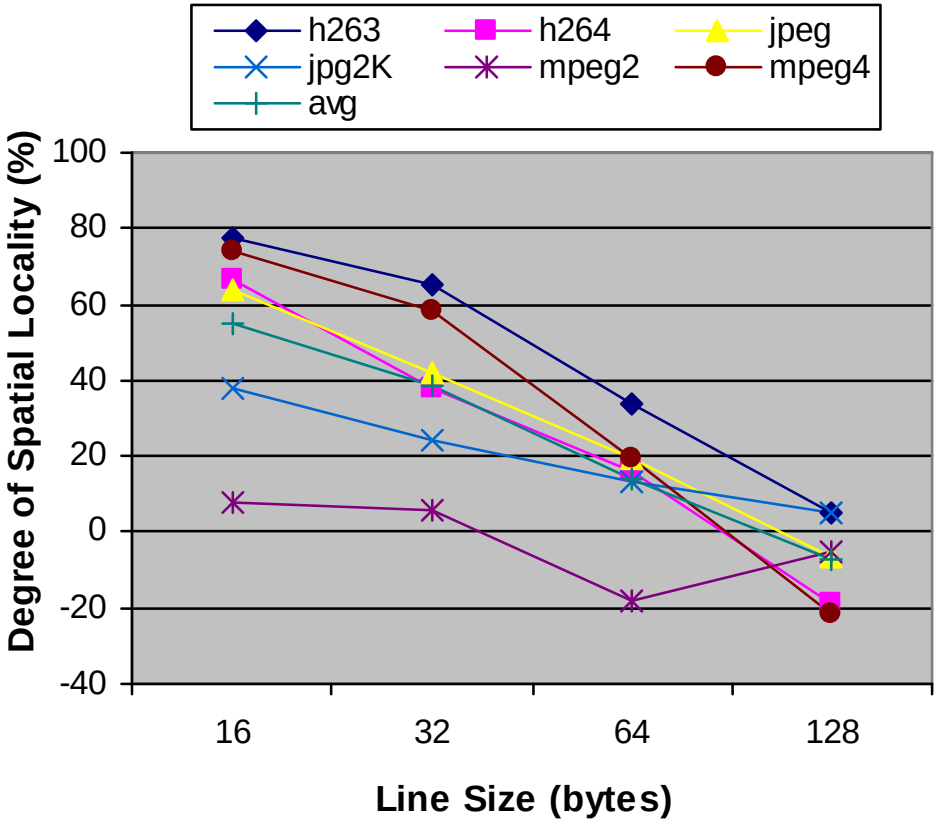
	Byte	Halfword	Word
MB video average	40%	35%	25%
SPECint2000	10%	5%	85%

Memory Statistics
working set size



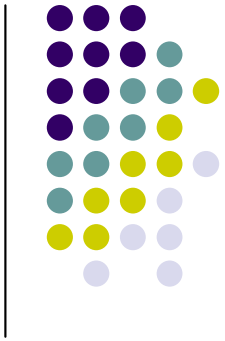
- The cache sizes requirements are fairly small
- 8KB of 2-way, 4-way and 8-way associate caches or 32 KB of direct-mapped cache are generally sufficient
- While general purpose:
  - 32K, 2 times worse
  - 128K for comparable results
  - Require often 2 times, often an order of magnitude larger cache sizes

Memory Statistics
spatial locality



- Spatial locality = (Mb-Ma)/(Mb/(la/lb))
- Best cache line size for data memory is 32 or 64 bytes in a 32KB cache
- balance the benefits of spatial locality from increasing line size vs. the caches conflicts from fewer sets
- while SPEC92: only small performance improvement from increasing line size

Memory Statistics
streaming data

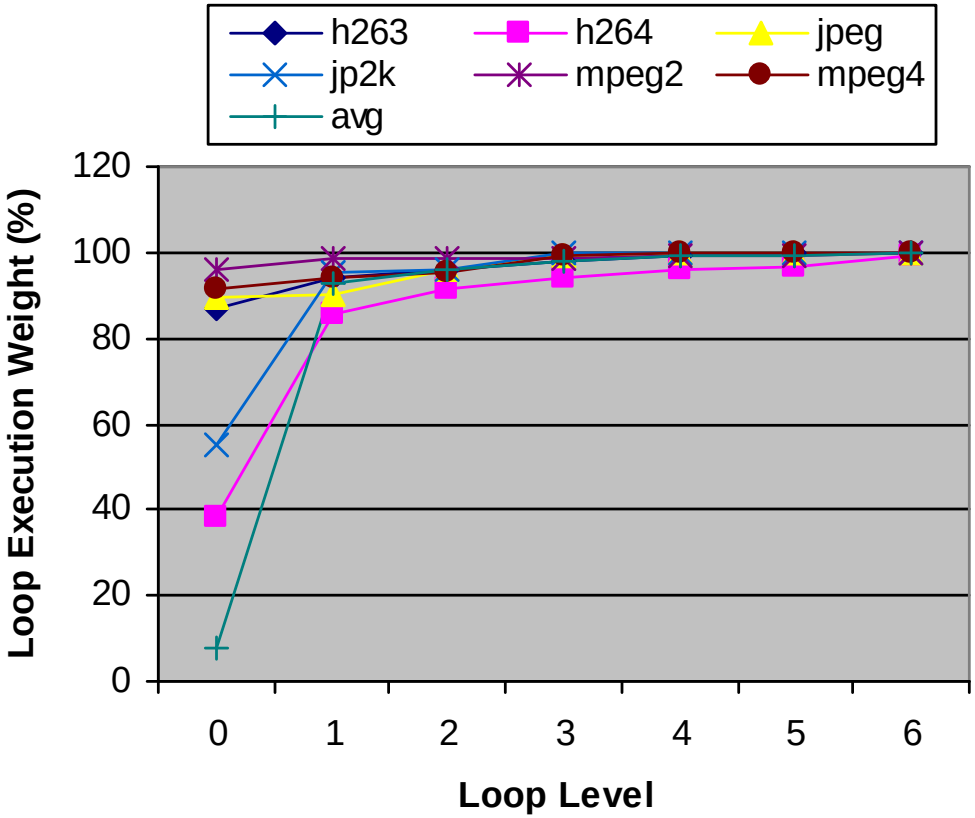


- Large amount of data
- Small working set size
- High frequency of memory access
- Good spatial locality

→ Nature of streaming data

→ Performance gain from memory prefetching support: stream buffer, stride prediction table, or a stream cache, etc.

Loop Statistics
loop level execution weight

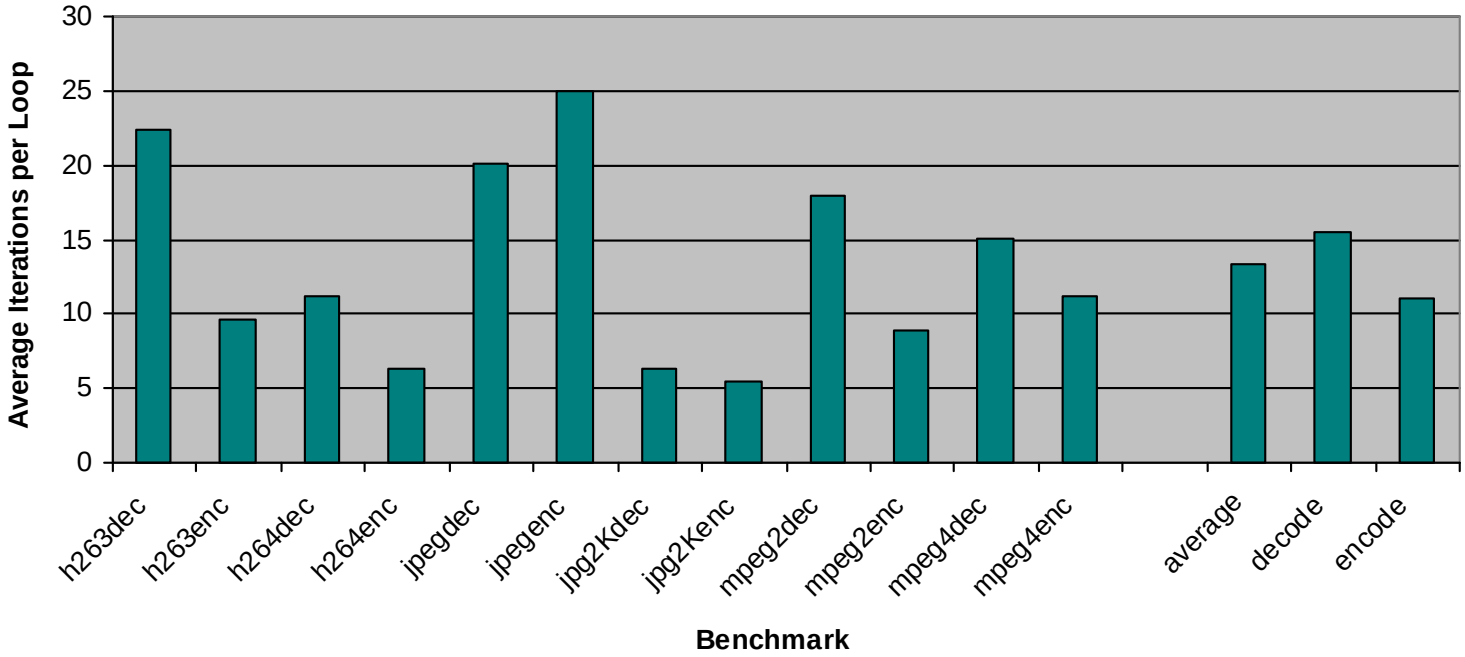


- Method of measure:

  depth-first search, assign higher weights to outer loops

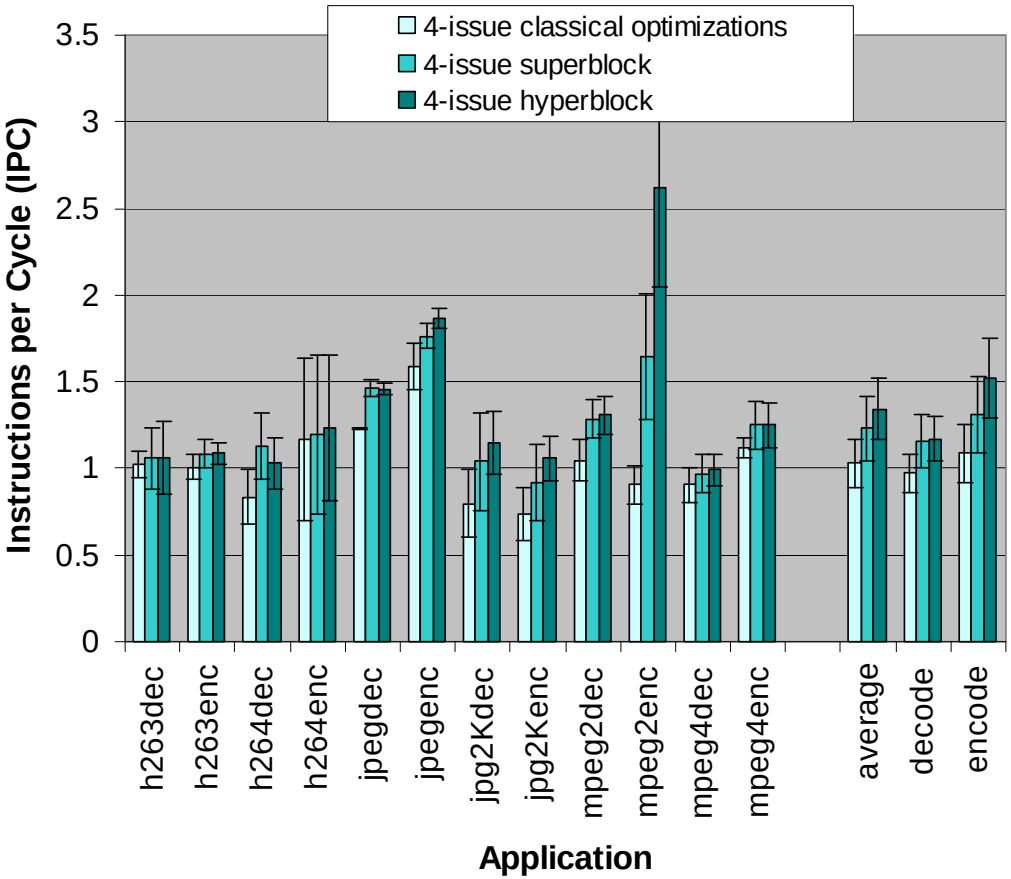
- 80-90% are inner loops
- 95% are two inner-most levels of loops
- higher processing regularity

Loop Statistics
loop iteration



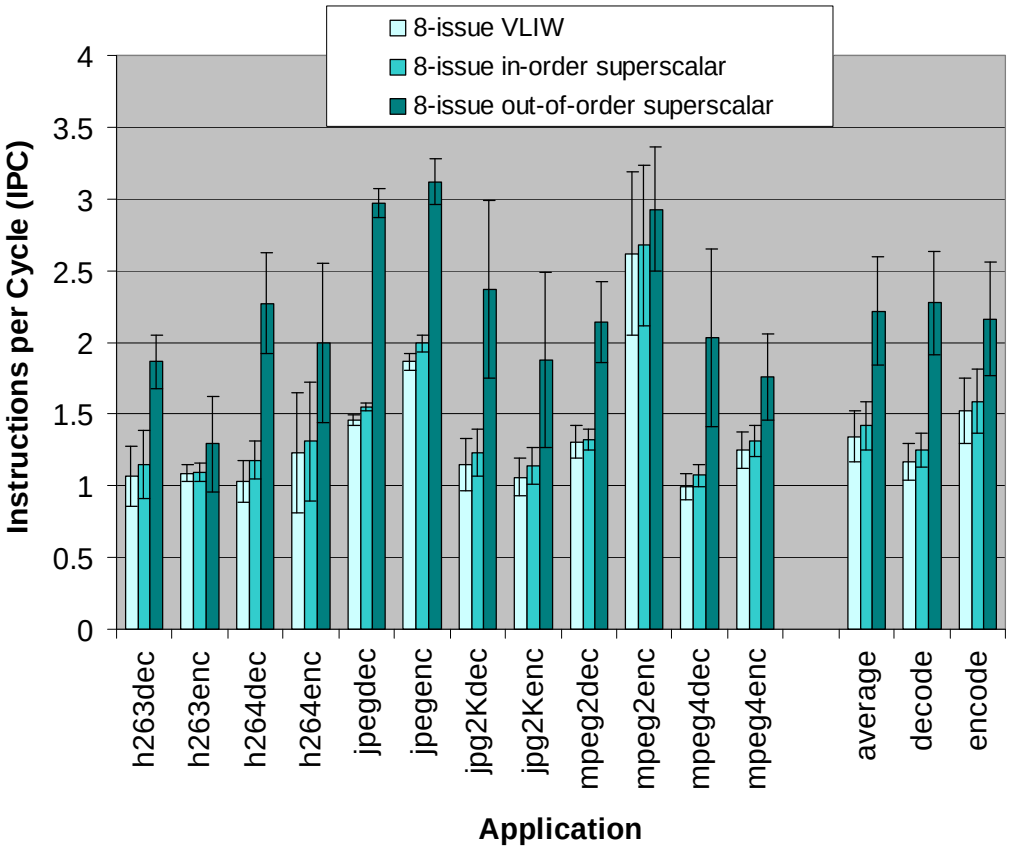
- Typical loops have a large amount of iterations
- video decoding entails more interactions per loop than encoding
- video applications are highly loop-oriented

Instruction Level Parallelism
static ILP



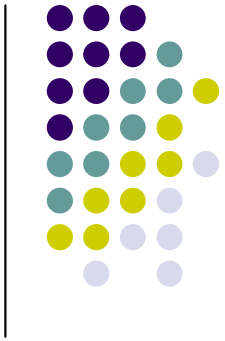
- Operation latencies
  - ALU instrs – 1 cycle
  - Loads – 2 cycles
  - Multiplies – 3 cycles
  - Floating-point – 3 cycles
  - Divides – 10 cycles
- Video applications contain a moderate degree of static ILP
- Greater ILP than general-purpose apps, but still modest
- Majority of parallelism in video is data parallelism, which is a coarser level of parallelism than ILP

Instruction Level Parallelism
static vs. dynamic ILP



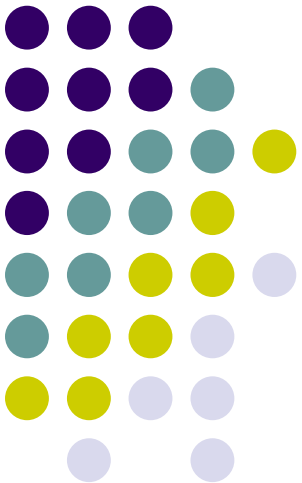
- Static scheduling performs nearly as well as dynamic in-order scheduling for media processing
- Dynamic out-of-order scheduling provides much better performance than static scheduling

Workload Variability Across Inputs

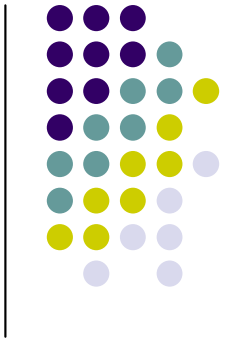


- When resolution of the video frames varies:
  - little variation was found
- When bitrate varies:
  - moderate effect on dynamic instruction count and L1 data cache miss rate
- When degree of motion varies:
  - higher degree of motion, greater execution time, dynamic instruction count increases
- When search window size varies:
  - has negligible impact on the workload characteristics of video decoding
  - significant variations in dynamic instruction count and L1 data miss rate for video encoding
- **Overall:**
  - other than execution time and cache miss rate, there was little impact on workload characteristics

Summary



Summary



- Introduction of MediaBench II Video

- Comprehensive workload evaluation of ***MBvideo***
  - instruction frequencies
  - basic block size
  - branch prediction rates
  - data size
  - working set size
  - spatial locality
  - loop characteristics
  - ILP scheduling performance

- MediaBench website: http://www.mediabench.org/
  - coming 1st quarter 2005