TI Developer Conference
February 28-March 2, 2006 • Dallas, TX

IMPLEMENTATION OF A 3-D FLEXIBLE MACROBLOCK ORDERING FOR H.264/AVC

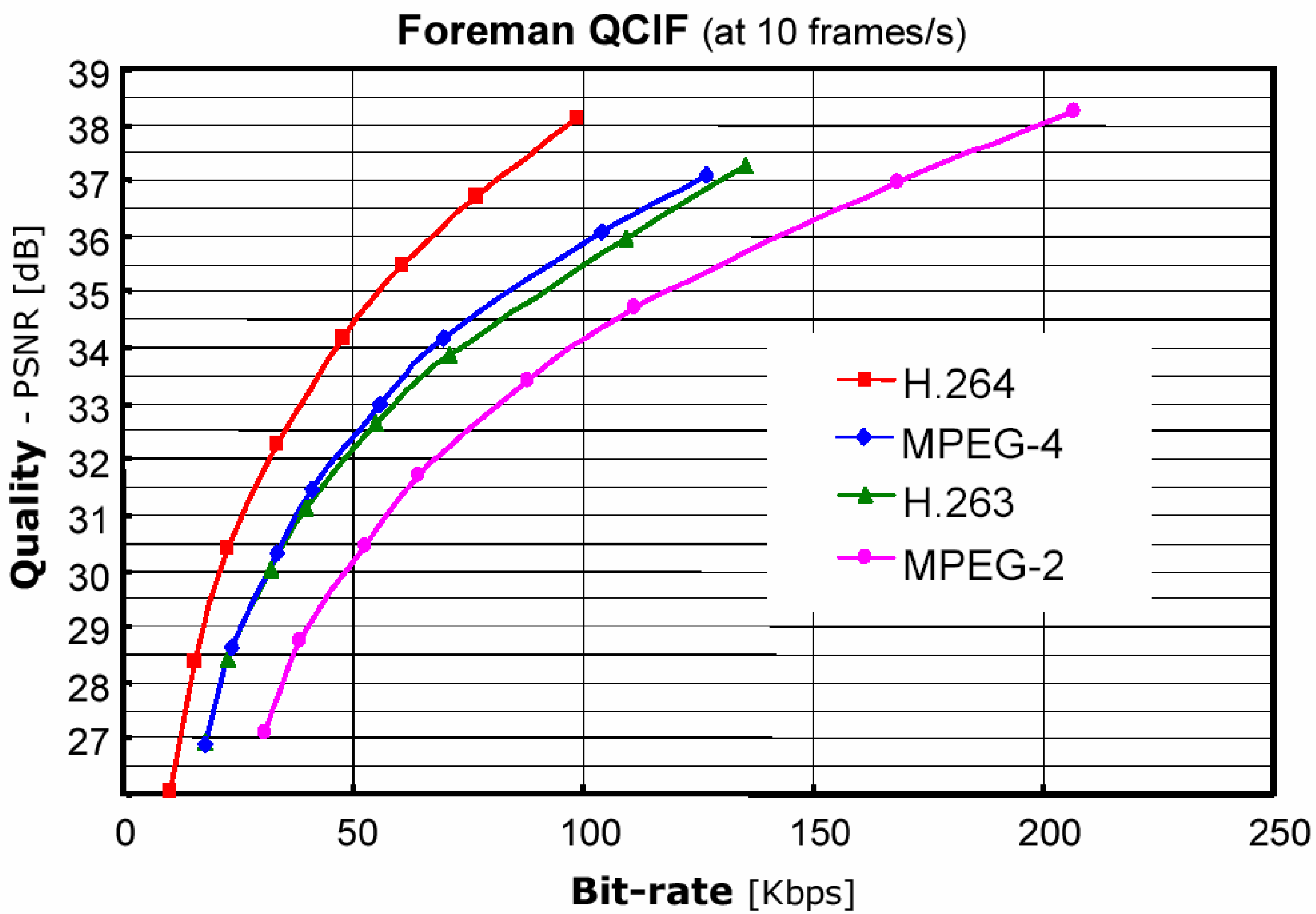
TOKUNBO OGUNFUNMI

Assoc. Professor
Santa Clara University
togunfunmi@scu.edu

OUTLINE

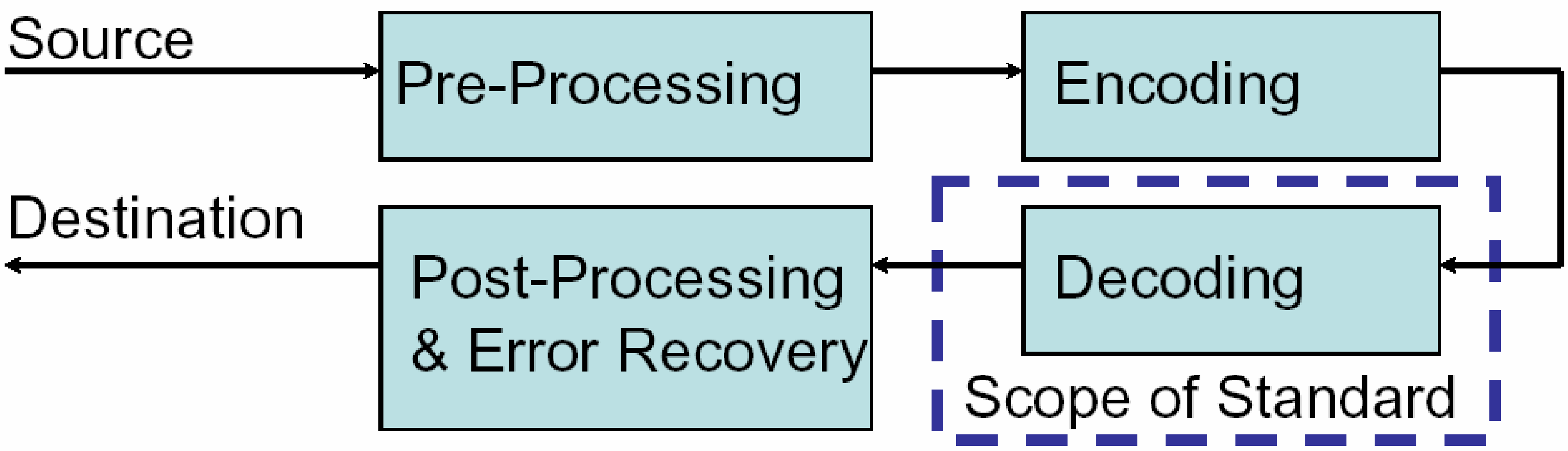
- Introduction
- 2D FMO
- 3D FMO Concepts
- Analysis of 3D FMO
- Conclusions

Comparison of Recent Standards

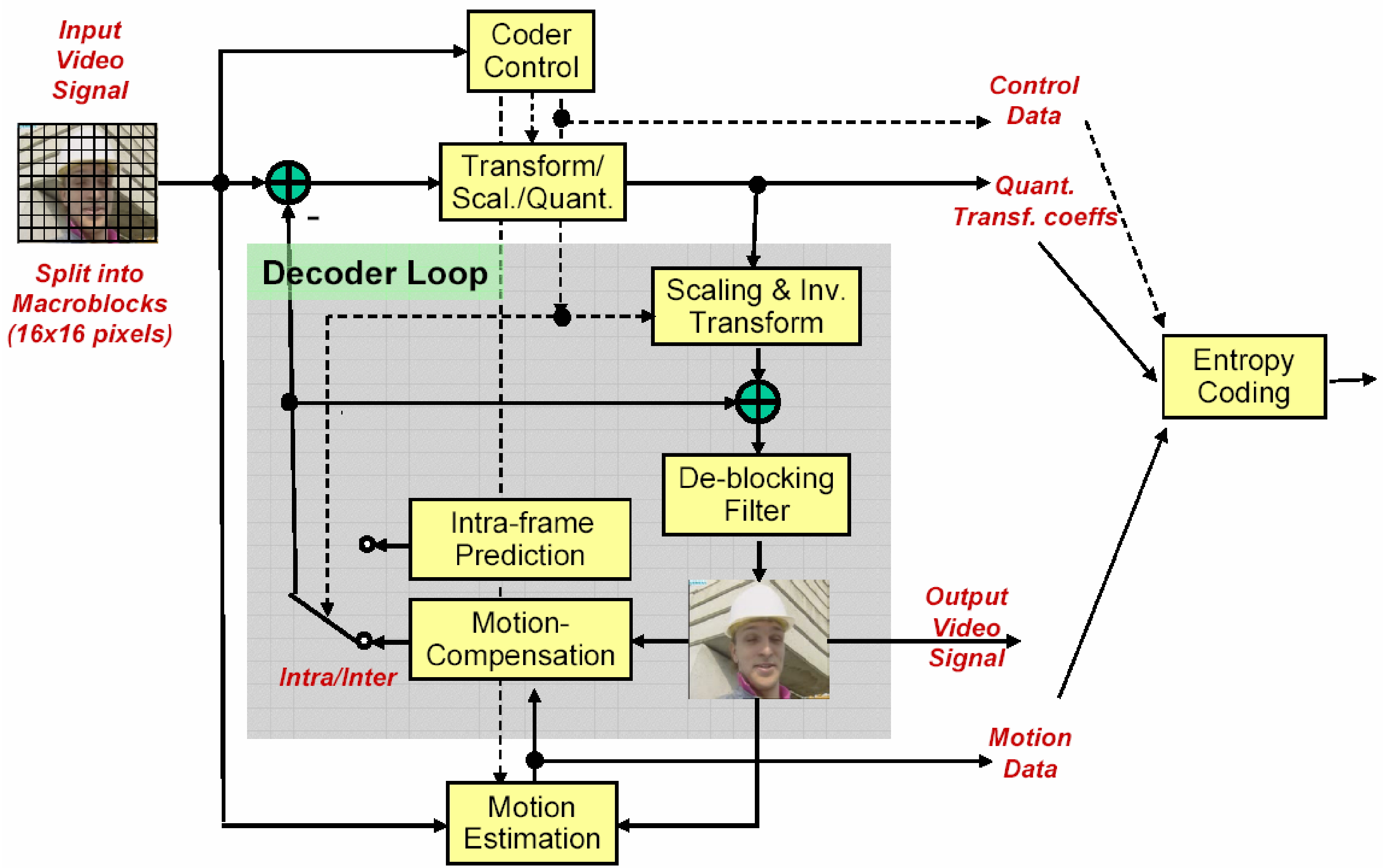


Scope of Picture and Video Coding Standardization

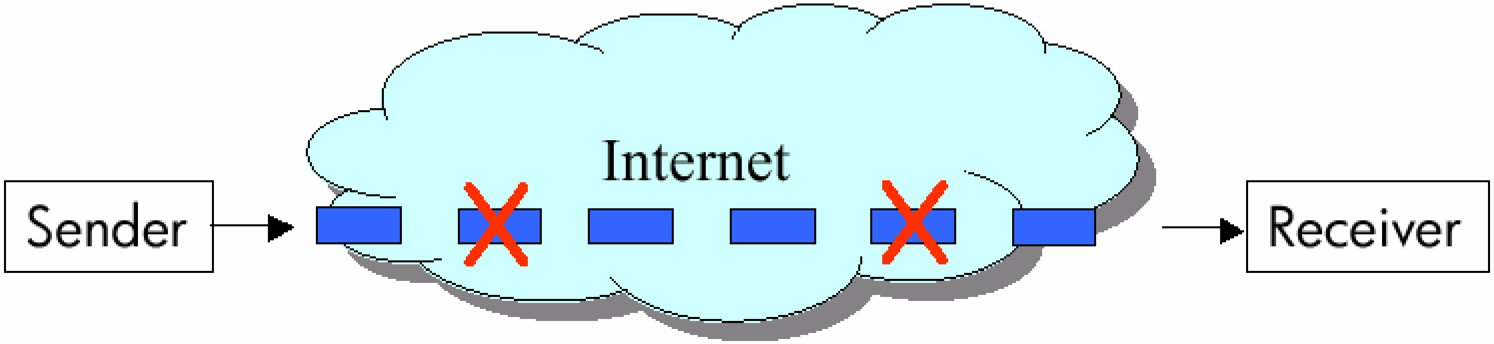
- Only the *Syntax* and *Decoder* are standardized:
  - Permits optimization beyond the obvious
  - Permits complexity reduction for implementability



Basic coding structure for H.264



Error Control



- Goal of error control:
  - Overcome the effect of errors, e.g. packet loss on a packet network or bit or burst errors on a wireless link
- Types of error control:
  - Channel Coding
    - Forward Error Correction (FEC)
    - Retransmission
  - Source Coding
    - Error concealment
    - Error-resilient video coding

Error resilient video coding

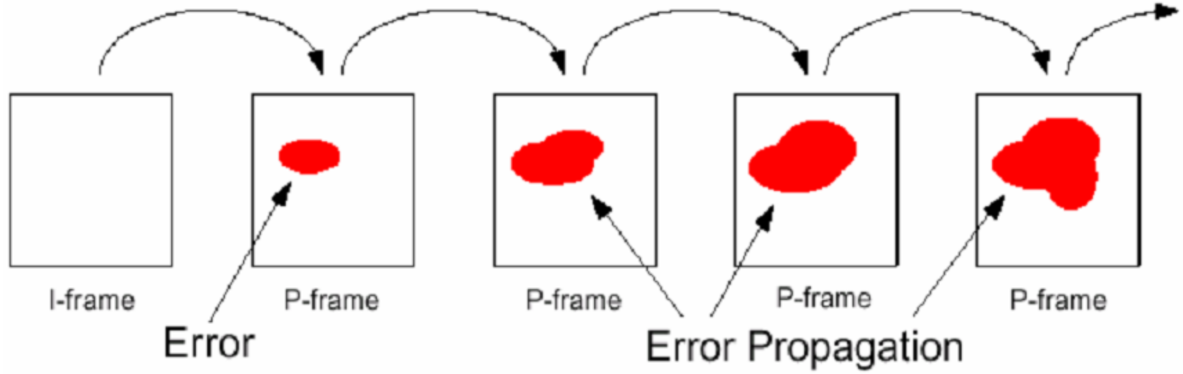
Two basic error-induced problems:

1. Loss of bitstream synchronization:

Decoder does not know what bits correspond to what parameters.

2. Incorrect state and error propagation:

Decoder's state (reconstructed frame) is different from encoder's, leading to incorrect predictions and error propagation



Error Resilience in H.264/AVC

Profile	Applications	Tools
Baseline	video conferencing & telephony/mobile applications	•Arbitrary Slice Ordering (ASO) to reduce the latency in real-time communication applications •Flexible Macroblock Ordering (FMO) •redundant slices
Main	broadcast video applications	enables an additional reduction in bandwidth over the Baseline profile : •bi-directional prediction (B-pictures) •Context-Based Adaptive Arithmetic Coding (CABAC) and weighted prediction.
Extended	streaming & mobile video applications.	Support all features except CABAC

Error Resilience Techniques in H.264/AVC

1. **Multiple Reference Frames**
2. **SP- and SI-Frames**
3. **Intra Picture Refresh & Intra Macroblock Refresh techniques.**
4. **Slice Structure coding, FMO**
5. **Etc.**

- **Some of these techniques are adopted from the previous video standards development experience.**
- FMO **is a** New Error Resilience Tool **for IP-Based Video and worth further investigation.**
- **It is implemented at both** encoder and decoder

FMO techniques....

.....improve error resilience in the coded video by

macroblock rearrangement

that distributes the error

throughout the frame which allows

error concealment

mechanism to conceal errors more effectively in the encoder.

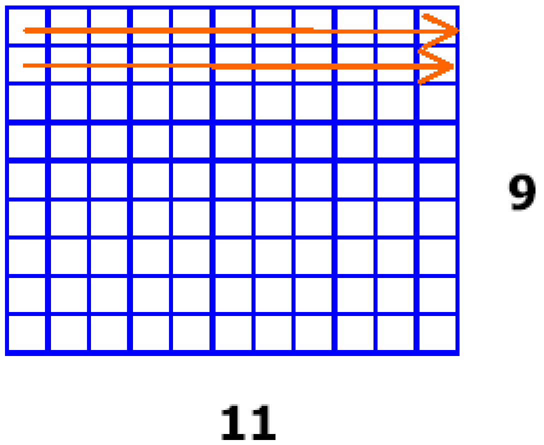
Video Coding Layer (VCL) of H.264

- ***Block-based*** **hybrid video coding approach (since H.263)**
- ***Macroblocks:*** **Each coded picture is represented in block shaped units of associated luma and chroma samples**
- **A coded picture contains either an entire *frame* or a single *field*.**
- **A picture maybe split into one or several *slices*.**

Organization of the Bit Stream

A given video picture is divided into a number of small blocks referred to as macroblocks.

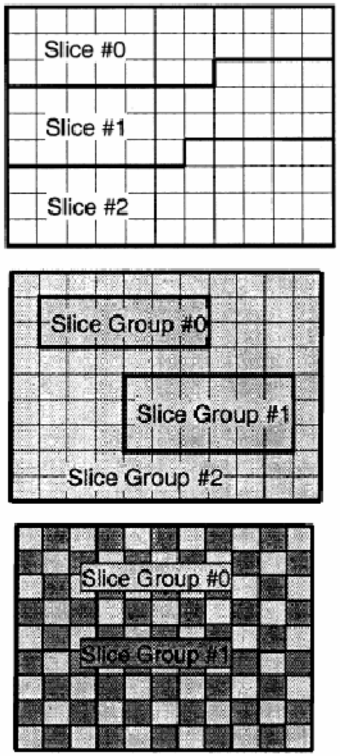
For example, a picture with QCIF resolution (176x144) is divided into 99 16x16 macroblocks as indicated in the following figure:



Subdivision of a QCIF picture into 16x16 macroblocks.

Slice and Slice Groups

- Slices are a sequence of macroblocks which is in raster scan order when not using FMO
- Slice are self-contained
- Flexible macroblock ordering (FMO) modified the way how pictures are partitioned into slices by utilizing slice groups

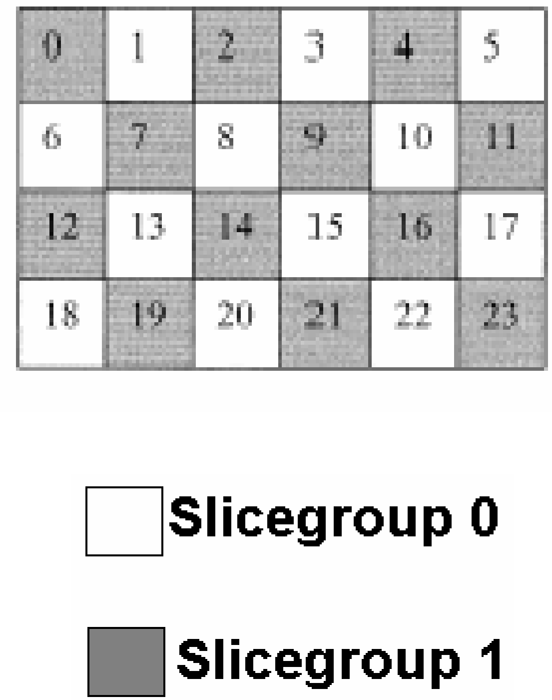


Slicegroups in FMO

- ❑ Each slice group can be partitioned into one or more slices.

- ❑ Each <u>slice group</u> is a set of macroblocks defined by a <u>*macroblock to slice group map*</u>.

- ❑ The <u>macroblock to slice group map</u> consists of a <u>slice group identification number</u> for each macroblock in the picture, specifying which slice group the associated macroblock belongs to.



Characteristics of Slice Structuring/FMO

- Slice structuring is a common error resiliency scheme used in many video compression standards.
- It stops spatial error propagation by limiting the error(s) inside a slice, which forbids any intraframe prediction across the slice boundary. The use of a smaller slice size will limit error propagation in smaller region, but the coding efficiency will decrease due to higher bit-rate overhead. Slice structuring aids error concealment as well.
- The use of FMO, aims at spreading the errors due to burst packet losses to the wider region of the frame. This makes the error concealment schemes more effective. But the use of checker board MB arrangement in FMO disallows the intra-frame prediction (to exploit spatial redundancy in neighboring blocks of a frame), which reduces the coding efficiency.

Scattered slices FMO

There are 7 different methods specified in the standard to convey an MBAmap (3 bits required per macroblock, 150 bytes for CIF(353x288 pixels)). This should be avoided. Therefore, there are 4 shortcuts :

"No FMO", "Scattered Slices", "Slice Interleaving", and "Fully flexible"

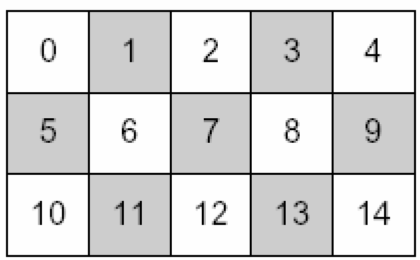


Figure 1: A graphic representation of an MBAmap. Depicted is a picture with 15 macroblocks numbered o to 14 in three macroblock lines. White macroblocks belong to SliceGroup 0, and gray macroblocks belong to SliceGroup 1.

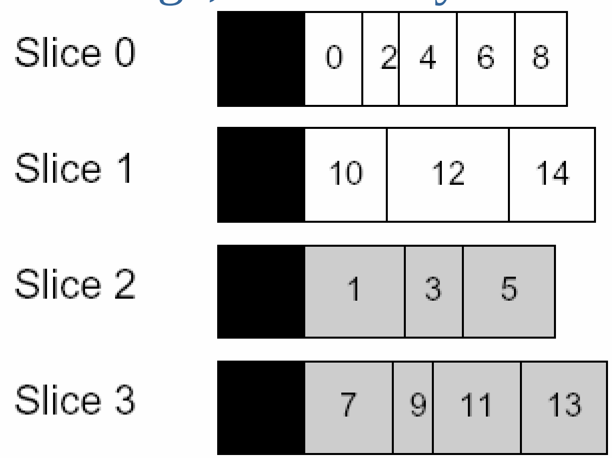


Figure 2: Coded representation of a picture, divided into two slice groups with two slices each. Black denotes the slice headers, and white and gray the macroblock data of macroblocks belonging to SliceGroup 0 and 1, respectively.

Reconstructed Macroblocks with FMO

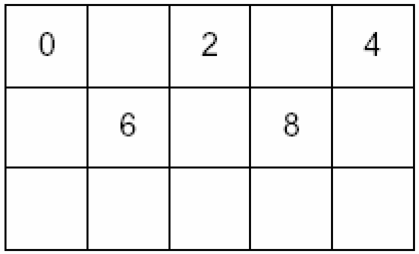


Figure 3: Reconstructed macroblocks of Slice 0

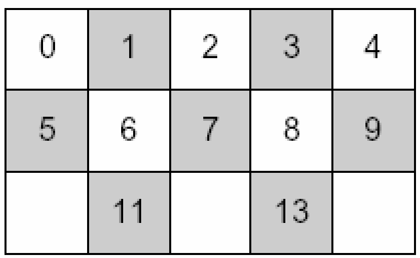


Figure 4: Reconstructed picture after an FMO slice loss. Note that the lost macroblocks are surrounded by reproduced macroblocks, which allows for efficient spatial and motion error concealment.

- Generally, the decoded picture will have "holes" in the locations of the missing macroblocks, but every hole has 4 neighbors that were decoded without error.
- A concealment algorithm may then leverage the correctly decoded information (which is spatially very close) to conceal the missing blocks.

3D FMO : How it works

- **Basic Ideas :**

1. Increase the number of usable (error-free) decoded neighboring macroblocks.
2. Improve the accuracy of the information of the neighboring macroblocks that are used for error concealment.

- **Why are these ideas important? :**

1. More useable macroblocks passed to error concealment tool at the decoder, the better error concealment result can be achieved.
2. During motion video, very often the macroblocks located at the previous and next frame are more relevant than the neighboring macroblocks on its current frame.

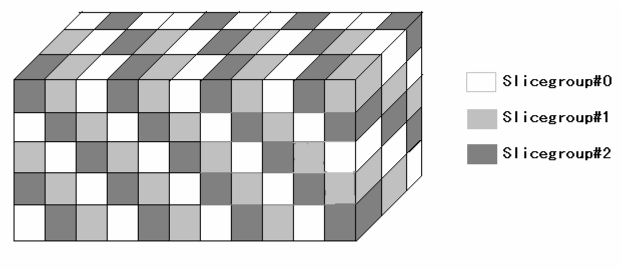
3D FMO : How it works

- **Implementation:**

The new FMO is implemented with a 3D Macroblocks allocation map (MBAmap) similar to a 3D magic cube. Therefore we call the new FMO technique 3D FMO. See Refs. [8], IEEE ISCAS Proceedings 2005 and [9] for details.

- **Result:**

A powerful error resilient tool, that outperformed the regular FMO especially under weak video transmission channels such as wireless networks.



3D FMO MBAmap

3D FMO Characteristics

- Encoder intelligently groups MBs into a slice whose size is less than (or equal to) the size of MTU (Maximum Transportation Unit). MTU is the largest size of a packet that can be transported through networks without being split.
- Prediction beyond the slice boundaries is forbidden to prevent error propagation from intra-frame predictions. The slice structuring strategy thus aims at avoiding error propagation from a corrupted packet to subsequent packets.
- The new 3D design will increase the number of useable neighboring macroblocks, therefore it will leave more valuable information for the decoder to conceal the error.
- The flexibility of 3D FMO allows various MBAmaps and ease of implementation.

3D FMO on Additional Memory

- The additional memory required is at the worst case, enough memory to store all the bits for two of the largest possible decoded frames. The reason for this is that given any three frames of the video sequence, you can reconstruct one complete frame using all the spatial and temporal neighboring macroblocks. Note that the additional memory required can be reduced by one frame buffer if macroblocks from only one temporal neighbor is used.

3D FMO on Additional Delay

- **The additional delay incurred is the delay involved in re-ordering the macroblocks before error concealment begins. This delay is also somewhat affected by packet arrival order although it is possible to reduce the effect of out-of-order packet arrival by specifying a time limit beyond which a packet is considered lost and therefore should be concealed. Given this mitigation, we consider only the delay incurred in macroblock re-ordering. In the worst case this is equivalent to the time it takes to search a one-dimensional array.**

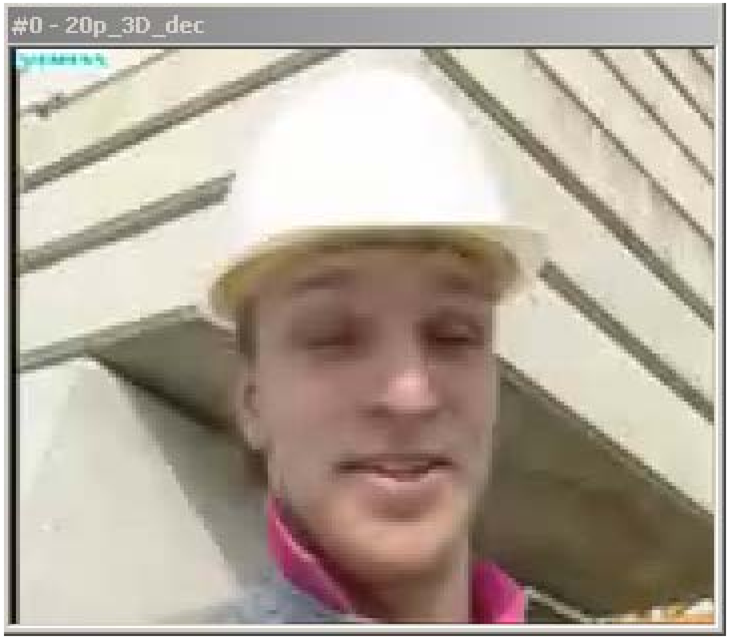
3D FMO on Coding efficiency

- **The abstraction of 3D FMO from the encoding (scan) order of the video sequence not only buys flexibility of application of the algorithm, but also no losses in the coding efficiency. This is because the algorithm is applied only after the sequence has been encoded. The original bitrate of the encoded sequence is retained.**

2D vs. 3D FMO

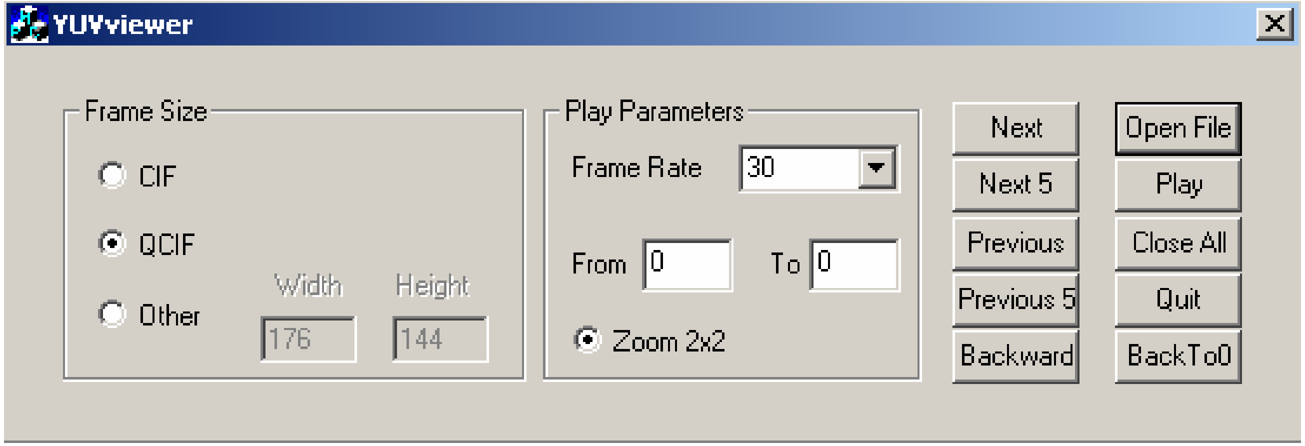
- **3D FMO will not introduce significant delay compared with 2D FMO in terms of coding efficiency.**
  - 3D FMO is interleaving the MB ID in the MBAmap (Macroblocks allocation map) level, not the MB itself.
    - Therefore, the memory consumed is saved.
    - It is actually a technique called “short cut” in FMO coding.
  - Like 2D FMO, 3D FMO is not creating new frames; therefore, it will not affect intra- or inter- coded picture sequences.
    - Therefore, the concern of reducing coding efficiency due to the broken intra- and inter-prediction mechanisms is unnecessary. However, this will not prevent error propagation from intra-frame predictions.

Test video, YUVviewer



Active type

Name	Format	Frame Rate	# of frames
Foreman	QCIF	10 fps	100



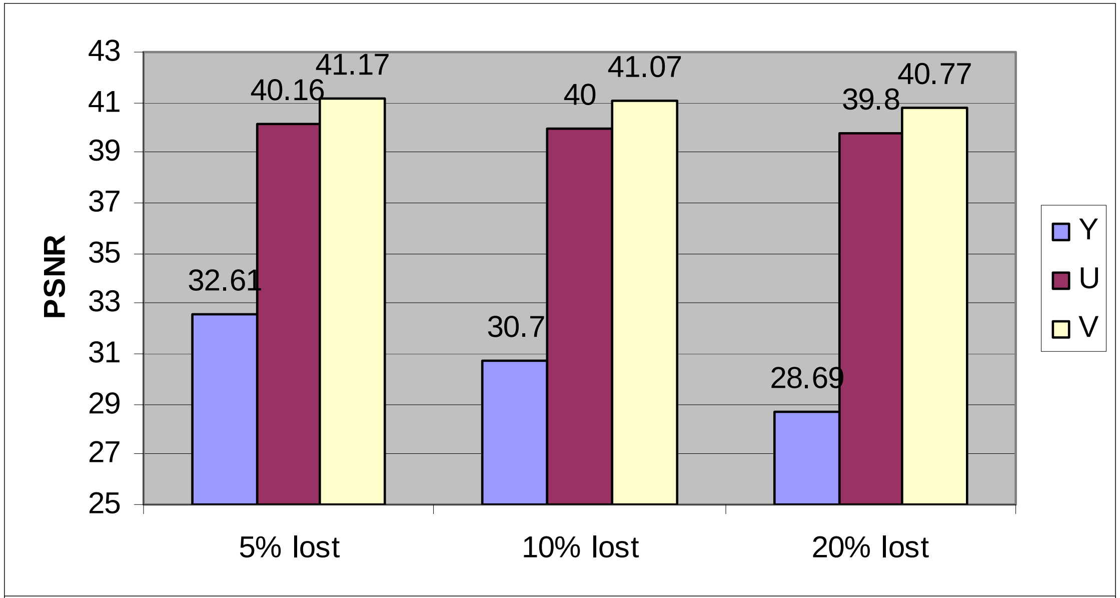
Comparison - subjective

Figure 1. Frame 3 of QCIF Foreman picture with 10% packets drop network **without FMO**

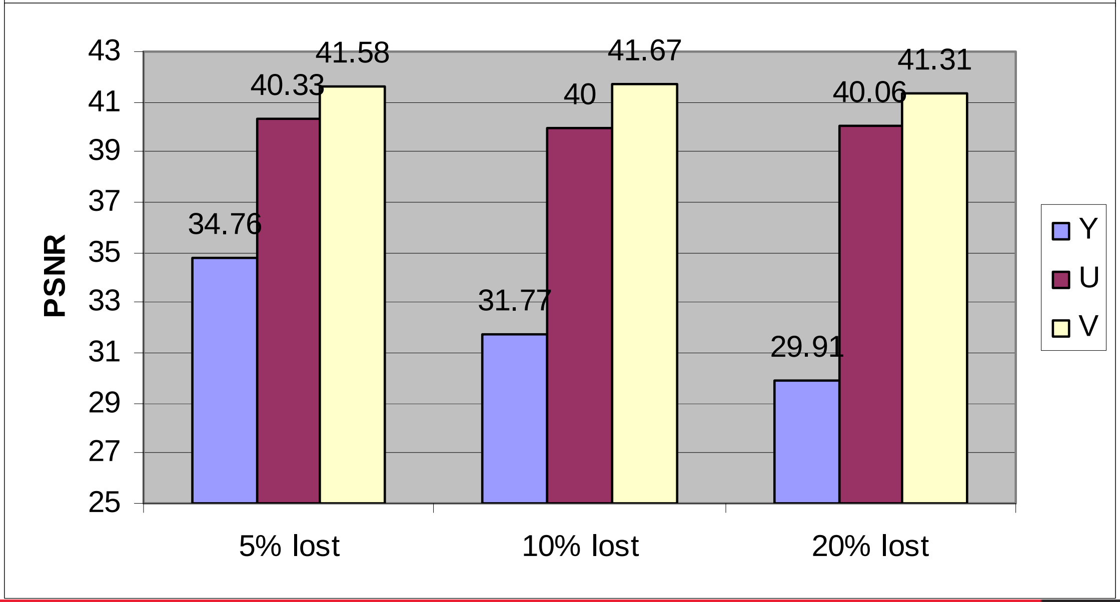
Figure 2. Frame 3 of QCIF Foreman picture with 10% packets drop network with traditional **scatter slice FMO**

Figure 3. Frame 3 of QCIF Foreman picture with 10%packet drop network with new **3D MBAmap FMO**

Comparison - objective

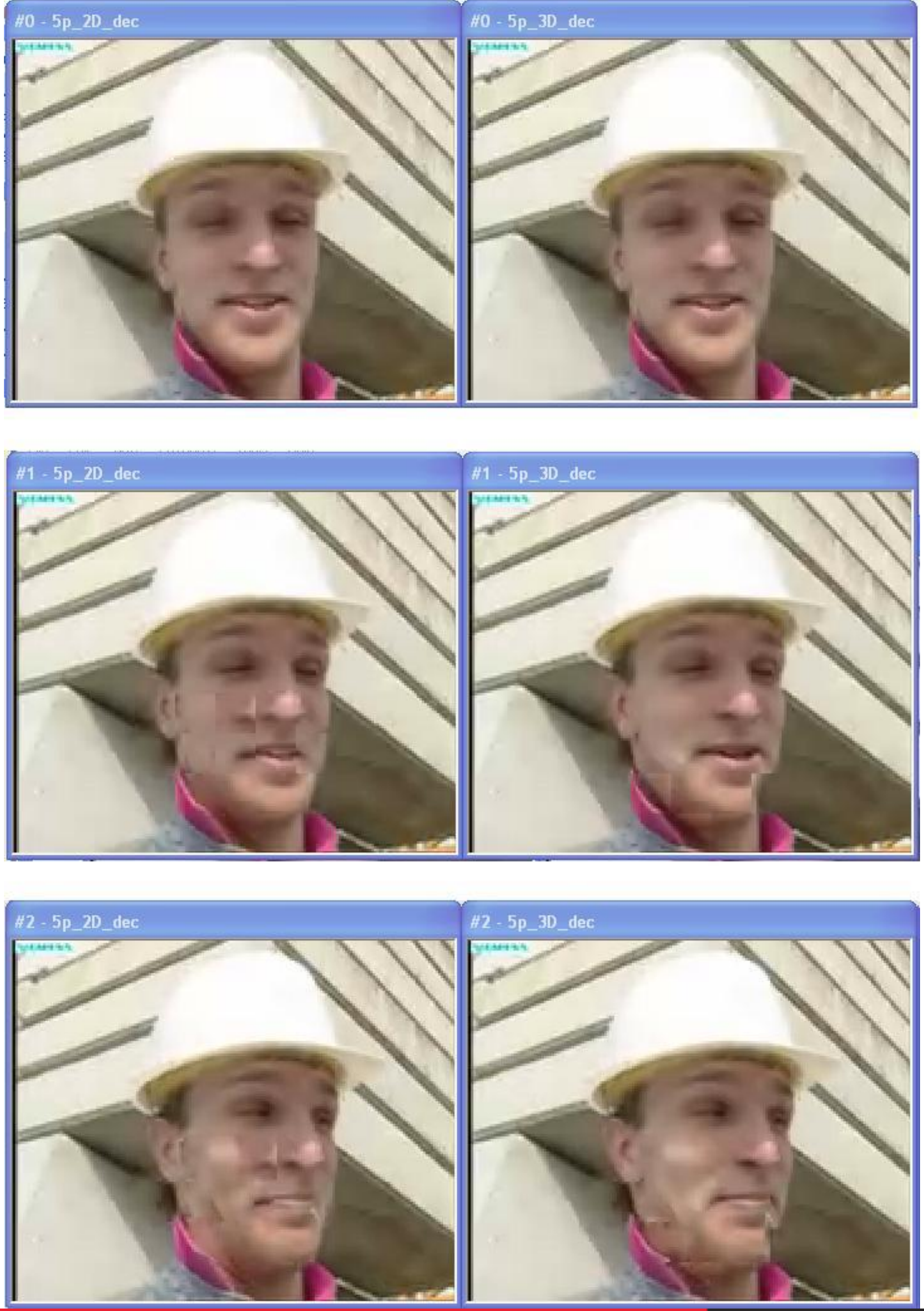


Foreman test frame 2 with regular FMO



Foreman test frame 2 with 3D FMO

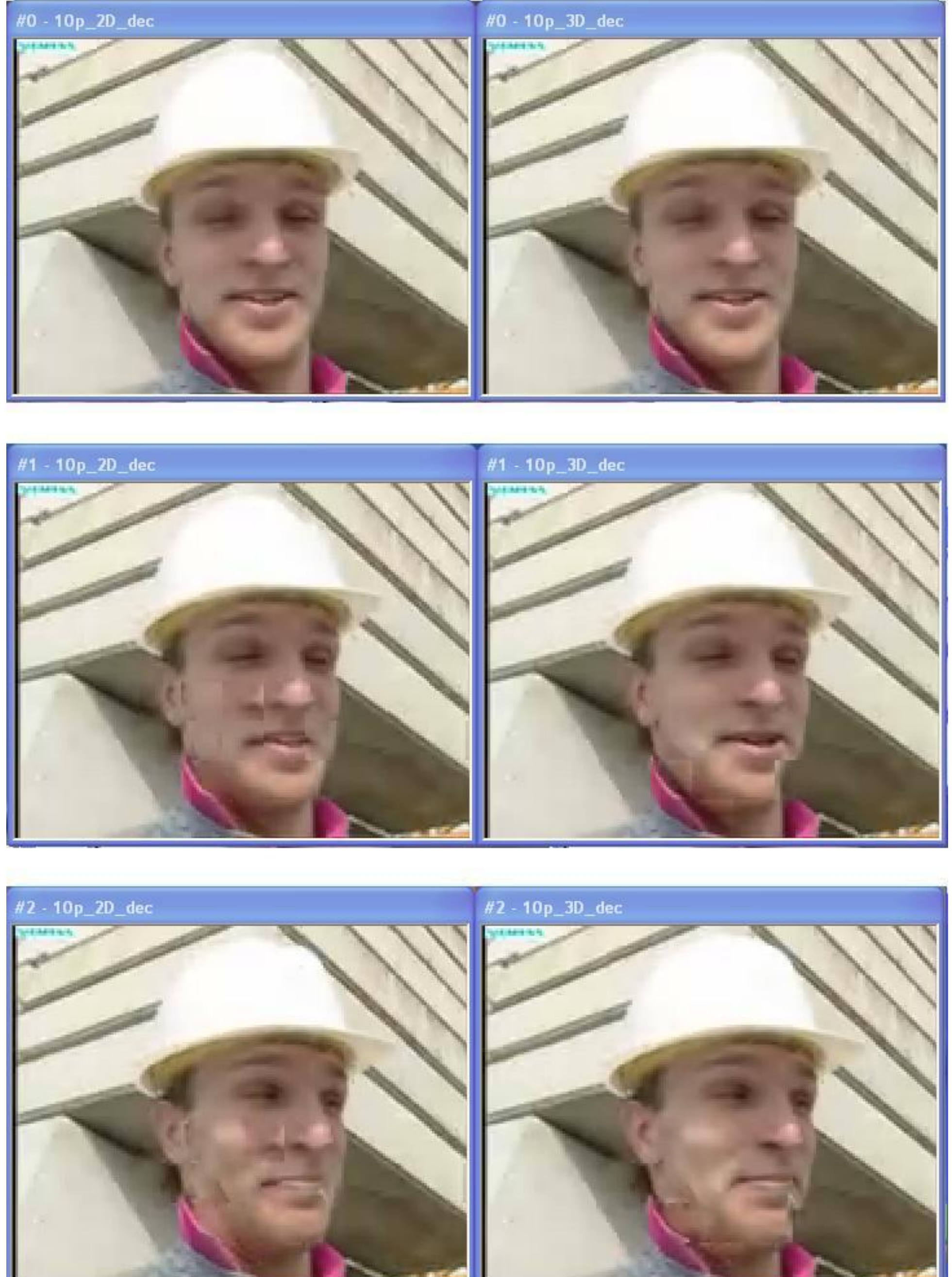
More test results(1)



Comparison between the conventional FMO and the 3D FMO each with first 3 frames of QCIF Foreman picture with 5% packet drop in

Wireless LAN

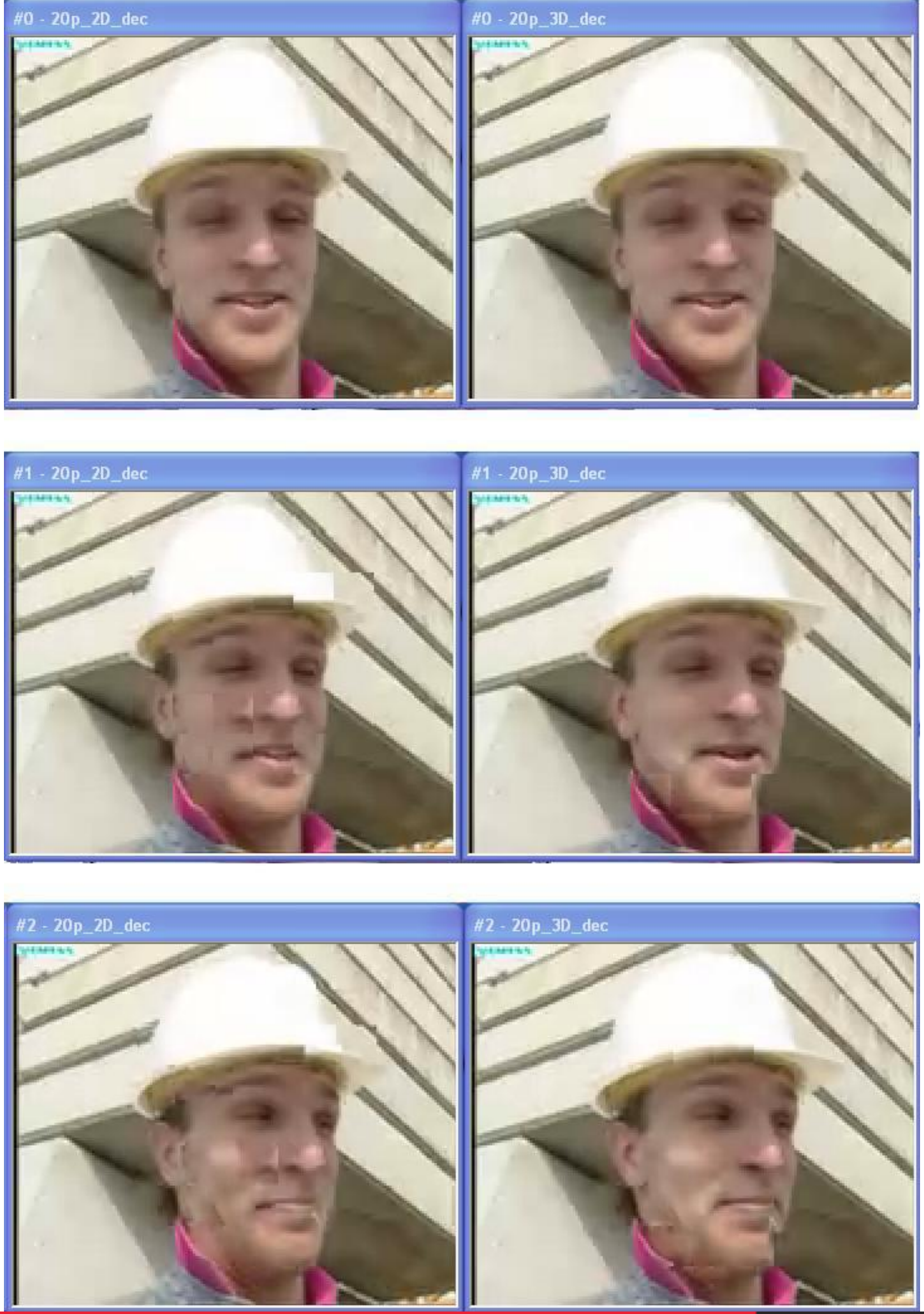
More test results(2)



Comparison between the conventional FMO and the 3D FMO each with first 3 frames of QCIF Foreman picture with 10% packet drop in Wireless LAN

More test results(3)

Comparison between the conventional FMO and the 3D FMO each with first 3 frames of QCIF Foreman picture with 20% packet drop WLAN



Conclusions

- We have introduced a new FMO technique that enhances the error resilient ability in the emerging H.264/AVC coding algorithms for transmission over lossy IP-based wireless networks. It is an extension of the regular FMO.
- The new FMO is a powerful error resilient tool. It outperforms the regular FMO especially under weak video transmission channels such as wireless networks.
- We analyzed the 3D FMO, found it suitable for environments where improved performance is desired at the cost of a modest expense on delay and a greater expense on latency. The target application suggested is video on demand over wireless networks.

References

- ✓ [1] Sullivan, G.; Wiegand, T.; Stockhammer T. "Using the Draft H.26L Video Coding Standard for Mobile Applications" - In: *IEEE International Conference on Image Processing 2001 (ICIP 2001), Thessaloniki, Greece*, October 2001
- ✓ [2] C.M. Calafate and M.P. Malumbres, "Testing the H.264 error-resilience on wireless ad-hoc networks", Proceedings of the EURASIP Video/Image Processing and Multimedia Communications Conference, 2003.
- ✓ [3] S. Wenger, "H.264/AVC over IP", IEEE transactions on Circuits and Systems for Video Technology. Vol. 13, No. 7, July 2003.
- ✓ [4] Thomas Wiegand, Gary J. Sullivan, Gisle Bjontegaard, and Ajay Luthra, "Overview of the H.264 / AVC Video Coding Standard," IEEE transactions on Circuits and Systems for Video Technology. Vol. 13, No. 7, July 2003.
- ✓ [5] S. Wenger, T. Stockhammer, and M.M. Hannuksela, "RTP payload Format for H.264 Video", draft-wenger-avt-rtp-h264-01.txt, Internet Draft, Work in Progress, March 2003
- ✓ [6] Karsten Sühring, "H.264/AVC Software Coordination," URL http://bs.hhi.de/~suehring/tml/
- ✓ [7] S. Wenger, M. Horowitz, *"Flexible Macroblock Ordering (FMO) 101,"* availablefromhttp://ftp3.itu.ch/av-arch/jvt-site/2002_07_Klagenfurt/JVT-D063.doc, July 2002.
- ✓ [8] T. Ogunfunmi, W. Huang, *"A Flexible Macroblock Ordering with 3D MBAmap for H.264/AVC,"* Proceedings of IEEE ISCAS May 2005.
- ✓ [9] William Huang*, "Error Resilient Video Coding for Wireless LAN Communications,"* MS Thesis, Departmentt. of Electrical Engineering, Santa Clara University, Santa Clara, CA 95053, April. 2004.
- ✓ [10]Y. Syed, *"FMO Cost Burden to Decoders Operating in Reliable Networks,"* available from http://ftp3.itu.ch/av-arch/jvt-site/2002_07_Klagenfurt/JVT-D121.doc, July 2002.

TI Developer Conference

Santa Clara University

IMPLEMENTATION OF A 3-D FLEXIBLE MACROBLOCK ORDERING FOR H.264/AVC

TOKUNBO OGUNFUNMI

Assoc. Professor
Santa Clara University
togunfunmi@scu.edu

SEE THE FUTURE

CREATE YOUR OWN