Optimization of Interactive Live Free Viewpoint Multiview Video Streaming Bandwidth

Richard Kramer, Member IEEE – Oregon State University

What if we could change *virtual* reality?...

into reality?

Optimization of Interactive Live Free Viewpoint Multiview Video Streaming Bandwidth

"Today, there are no known streaming services that provide MVV [Multi-View Video] content to home users …

… [because it is] infeasible to perform transmission over fixed bit-rate channels …" [Dufaux2013]

Richard Kramer, Member IEEE – Oregon State University

What if we could change *virtual* reality?... *into reality?*

Agenda

- Background
  - Video Compression and SVC (Scalable Video Coding)
- Multiview Video Types
  - Multiview Coding (MVC) Types and Industry Standards
- State of the Industry – FTV (Free Viewpoint TV)
- OLFVmv (Optimized Live Free Viewpoint multiview video)
  - Motivation
  - Contribution
  - Architecture
  - Algorithms
- Simulation Results
- Extensions to OLFVmv Using Network Coding
- Further Optimization of OLFVmv – Ph.D. Thesis Work
- Conclusion

Agenda

- Background
  - Video Compression and SVC (Scalable Video Coding)
- Multiview Video Types
  - Multiview Coding (MVC) Types and Industry Standards
- State of the Industry – FTV (Free Viewpoint TV)
- OLFVmv (Optimized Live Free Viewpoint multiview video)
  - Motivation
  - Contribution
  - Architecture
  - Algorithms
- Simulation Results
- Extensions to OLFVmv Using Network Coding
- Further Optimization of OLFVmv – Ph.D. Thesis Work
- Conclusion

MPEG Video Compression Building Blocks

Video Compression Steps

Step 1: Reduction of Resolution

Step 2: Motion Estimation

Step 3: Discrete Cosine Transform (DCT)

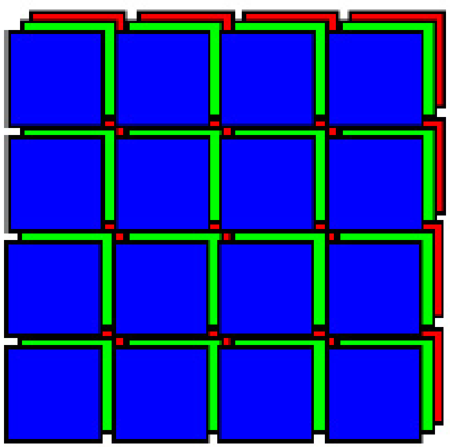
Step 4: Quantization

Step 5: Entropy Coding

MPEG Video Compression Building Blocks

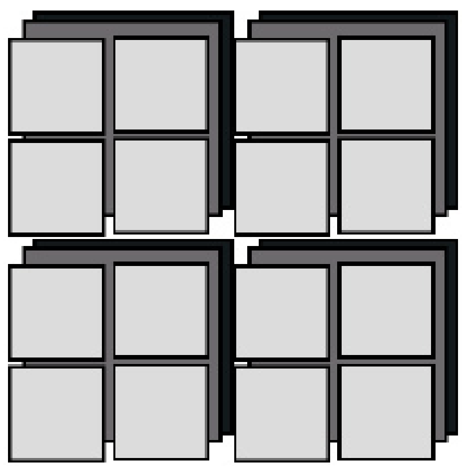
Step 1: Reduction of Resolution

Visual perception of color space (U and V) is much lower than to Luminance (Y). Thus color information for U and V can be combined more so than for Y.



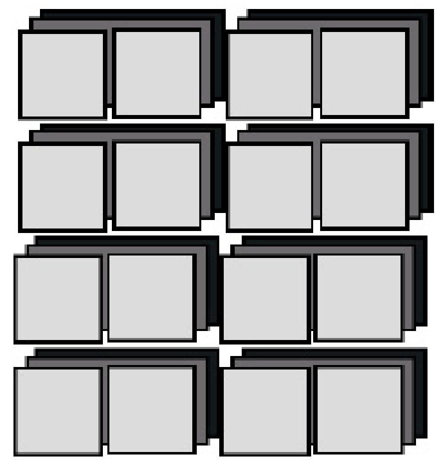
RGB

16 Discrete Pixels=100%



YUV (4:2:0)

Same Y, 1/4U and 1/4 V =50%



YUV (4:2:2)

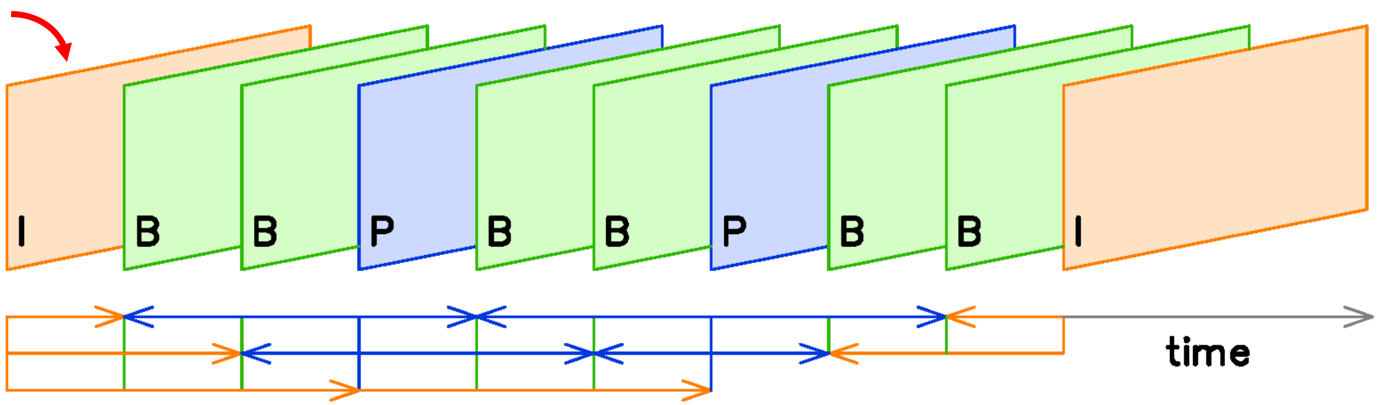
Same Y, 1/2U and 1/2V = 33%

[Mitrovic]

MPEG Video Compression Building Blocks

Step 2: Motion Estimation

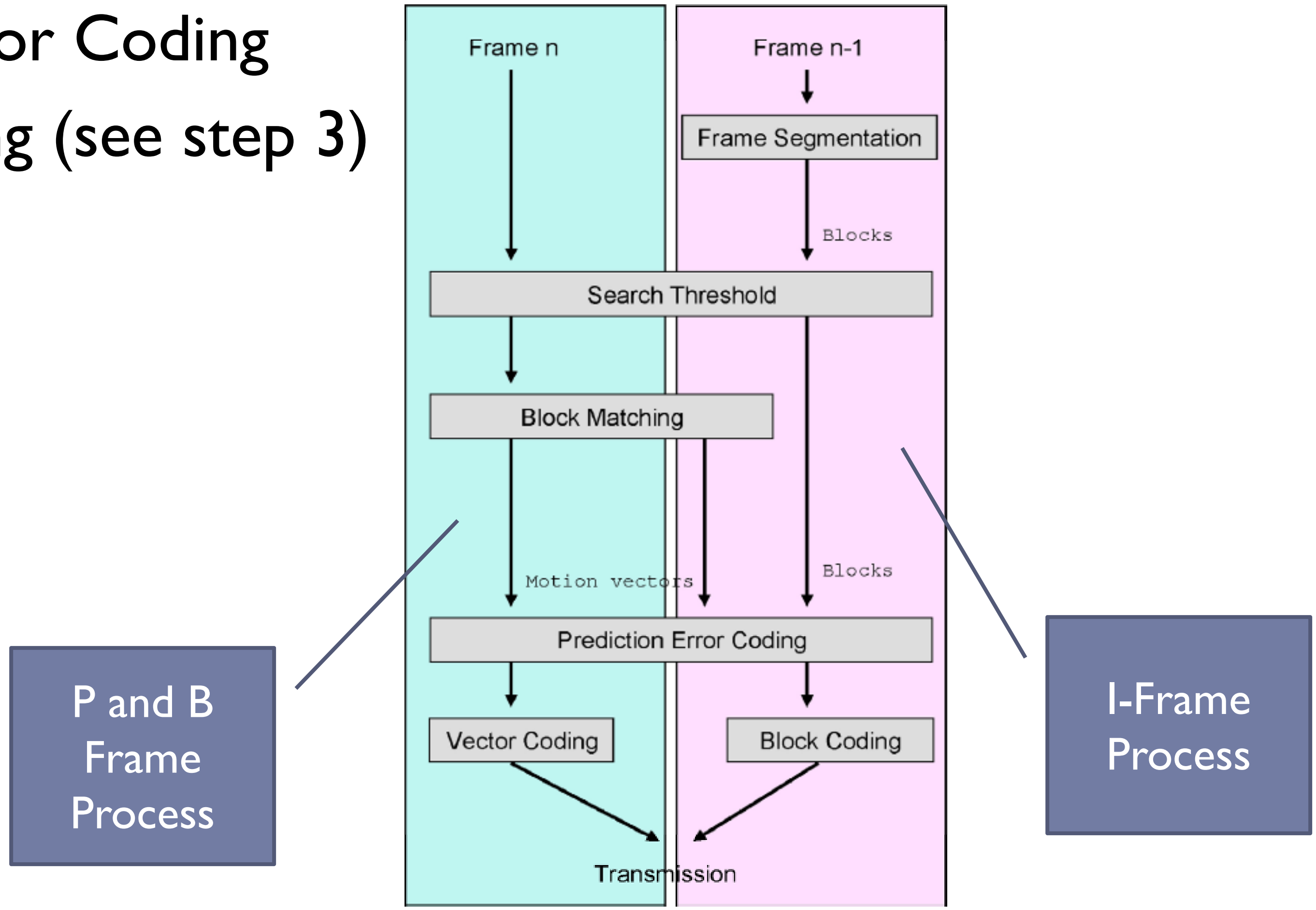
- MPEG employs multiple Frame Types
  - I (Intra) Frames – Spatially encoding of entire image
  - P (Prediction or Inter-Predication) Frames – Uses information from <u>ONE</u> reference point in time to create an image
  - B (Bi-Directional) Frames - Uses information from <u>TWO</u> reference points in time to create an image



MPEG Video Compression Building Blocks

Step 2: Motion Estimation – entails numerous sub-steps

- Motion Vector Coding
- Block Coding (see step 3)

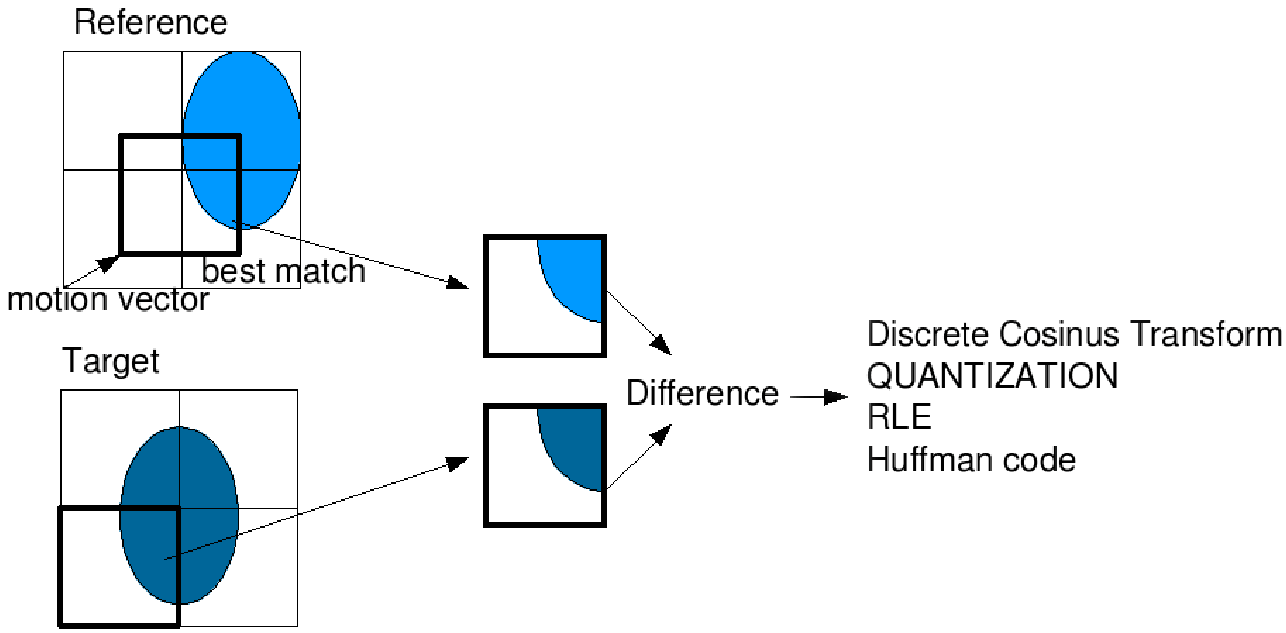


[Mitrovic]

MPEG Video Compression Building Blocks

Step 2: Motion Estimation – entails numerous sub-steps

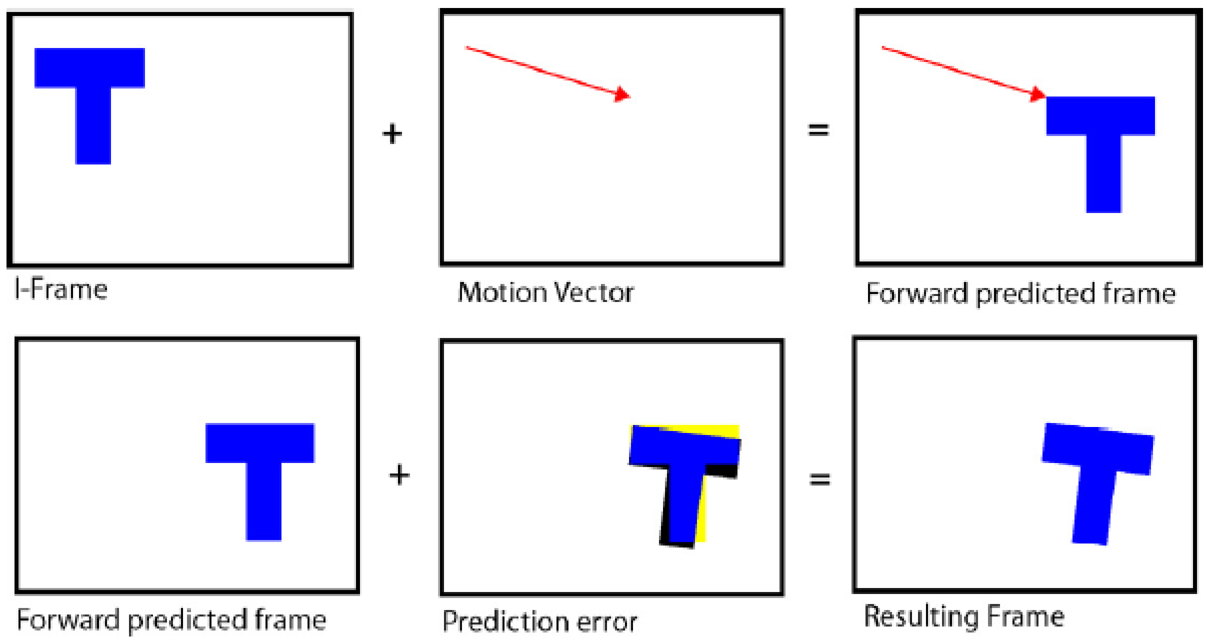
- Block Matching
  - A Block Matching Algorithm is used to look at the surrounding macroblocks to see if there is a match to the "Reference" macroblock



MPEG Video Compression Building Blocks

Step 2: Motion Estimation – entails numerous sub-steps

- Motion Vector and Error Correction
  - Once the matching macroblock is found and the correction is evaluated, a motion vector is generated identifying where to move the "Reference" macroblock + Error Correction

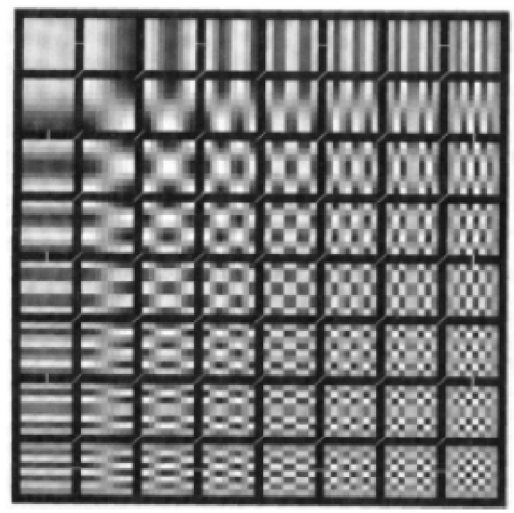


[Mitrovic]

MPEG Video Compression Building Blocks

Step 3: Discrete Cosine Transform (DCT)

- Each macroblock is analyzed to determine the contribution of EACH of the below 64 visual "frequencies".
- The associate "weights" of each of the 64 DCT possible frequencies are called the "DCT coefficients"



[Mitrovic]

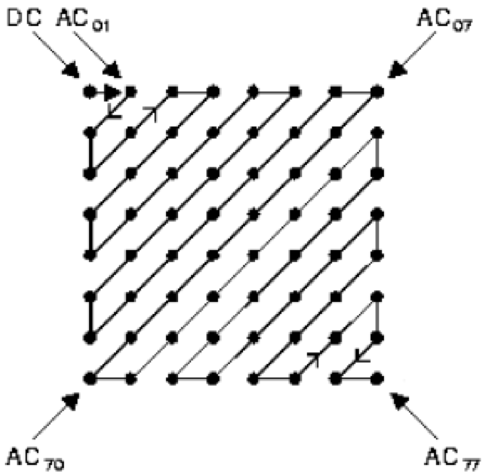
Oregon State UNIVERSITY OSU

MPEG Video Compression Building Blocks

Step 4: Quantization

Based on the desired quality level, the 64 DCT coefficients are then additionally scaled based on human visual perception, e.g., higher frequency components are less noticeable to humans thus are given less weight (or set to zero)

The results of the 64 quantized DCT coefficients are then stored in a zig-zap pattern



[Mitrovic]

Oregon State University OSU

MPEG Video Compression Building Blocks

Step 5: Entropy Coding

The DCT differentials are then calculated using variable-length codes to obtain further compression.

Category C	Range of *DIFF* value	Example codeword
0	0	00
1	-1, 1	010
2	-3, -2, 2, 3	011
3	-7..-4, 4..7	100
4	-15..-8, 8..15	101
5	-31..-16, 16..31	110
6	-63..-32, 32..63	1110
7	-127..-64, 64..127	11110
8	-255..-128, 128..255	111110
9	-511..-256, 256..511	1111110
10	-1023..-512, 512..1023	11111110
11	-2047..-1024, 1024..2047	111111110

[Schwarz2013]

Agenda

- Background
  - Video Compression and SVC (Scalable Video Coding)
- Multiview Video Types
  - Multiview Coding (MVC) Types and Industry Standards
- State of the Industry – FTV (Free Viewpoint TV)
- OLFVmv (Optimized Live Free Viewpoint multiview video)
  - Motivation
  - Contribution
  - Architecture
  - Algorithms
- Simulation Results
- Extensions to OLFVmv Using Network Coding
- Further Optimization of OLFVmv – Ph.D. Thesis Work
- Conclusion

Multiview Video Types

The H.264/MPEG4 MVC (Multiview Coding) Standard was first approved in July 2008

- Integrated into 5th Edition of H.264/MPEG-4 Std. ISO/IEC 14496-10 (Annex H)

Two specific Multiview "Profiles" are supported:

1) Stereo High Profile, also known as "3D" or "2D plus Delta"
   - Used for 3D movies including Blue-Ray
   - Various methods are employed to display 3D movies (glasses, holographic displays, etc.

2) Multiview High Profile supports an arbitrary number of views, also know as "Free-viewpoint Vide o" or "FTV" (Free-viewpoint TV)

- FTV is used for example, to obtain differing views of a field in a sports competition, such as soccer.

Important H.264/MPEG4 Revisions:

Version 11: (March 16, 2009) Major addition to H.264/AVC containing the amendment for Multiview Video Coding (MVC) extension, including the **Multiview High profile**.

Version 12: (March 9, 2010) Amendment containing definition the **Multiview Stereo High profile** for two-view video coding with support of interlaced coding tools and specifying an additional SEI message (the frame packing arrangement SEI message).

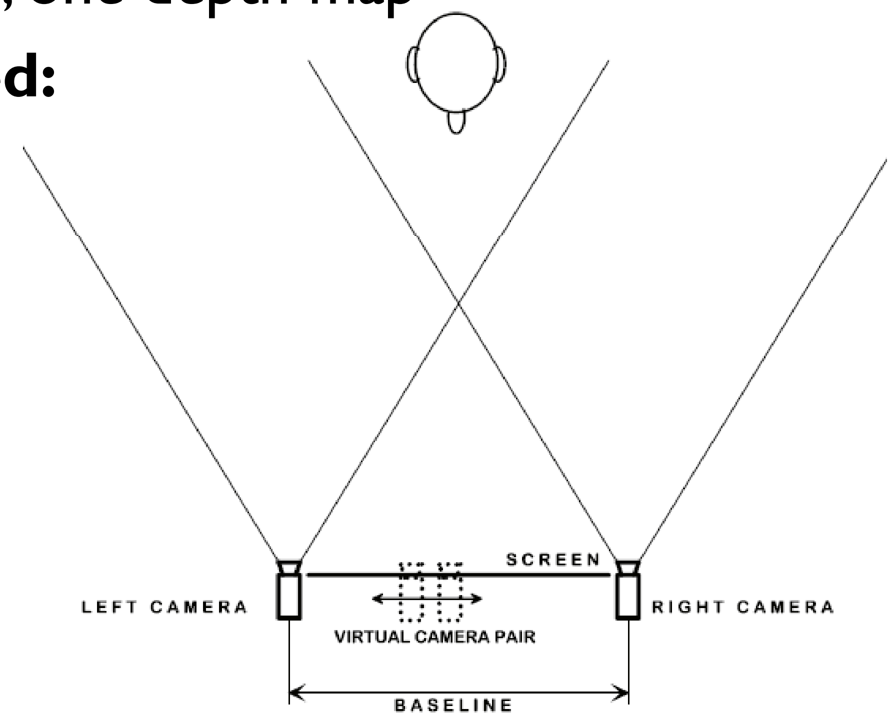
Version 18: (April 13, 2013) Amendment to specify the coding of depth map data for 3D stereoscopic video, including a **Multiview Depth High profile**.

[ISO/IEC 14496-10:2008][ISO/IEC 14496-10:2009][ISO/IEC 14496-10:2010][ISO/IEC 14496-10:2014]

1) Stereo/3D Multiview Video

Typically two (2) cameras, the primary view and associated depth map(s) is encoded

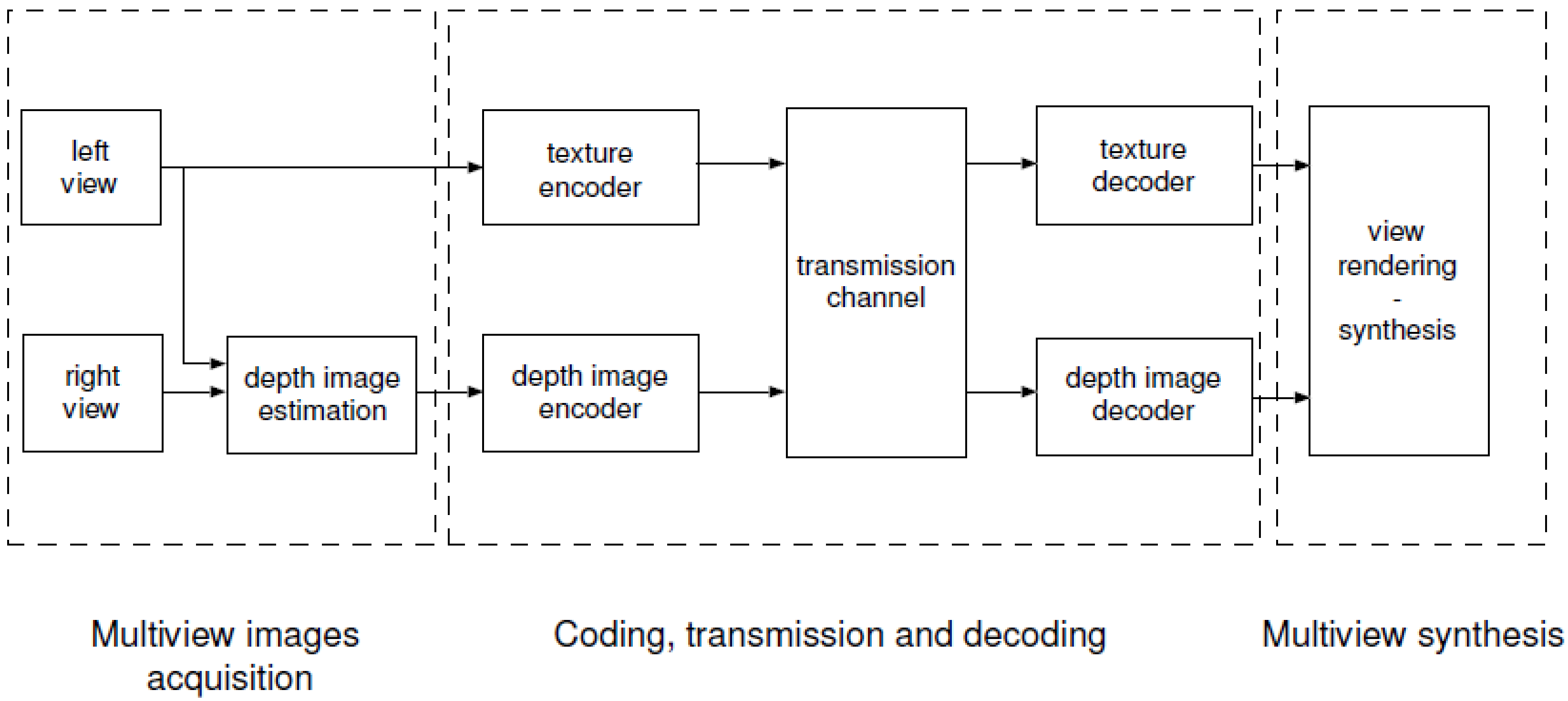
- Generate synthesized views using video and depth
- At minimum: One video, one depth map
- **Technologies required:**
  - Depth estimation
  - Depth encoding
  - View synthesis



[Ohm2009]

1) Stereo/3D Multiview Video

For MVC (Multiview Coding), a “based frame” is used (for example, the “left view” and relative to that, and only prediction information is transmitted relative to the “right view”



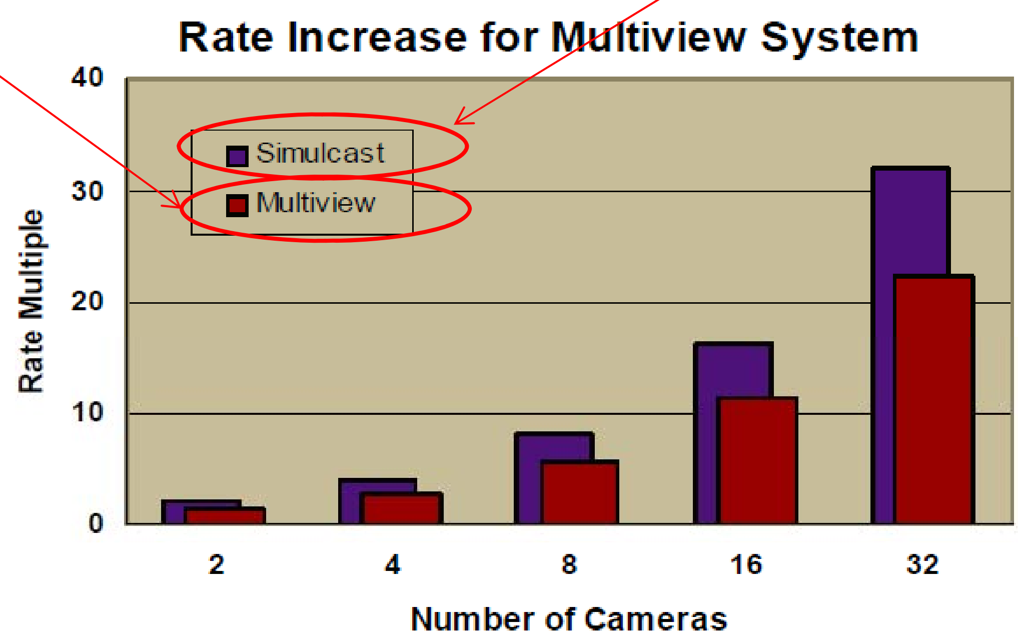
[Morvan_deWithFarin2006]

2) High Profile / FTV Multiview Video

- Multiple cameras whereas what is displayed is either part of an actual image, or a synthetic image, created by a combination of other images.

e.g., if all camera images including their P-Frame/B-Frame interdependencies are sent together

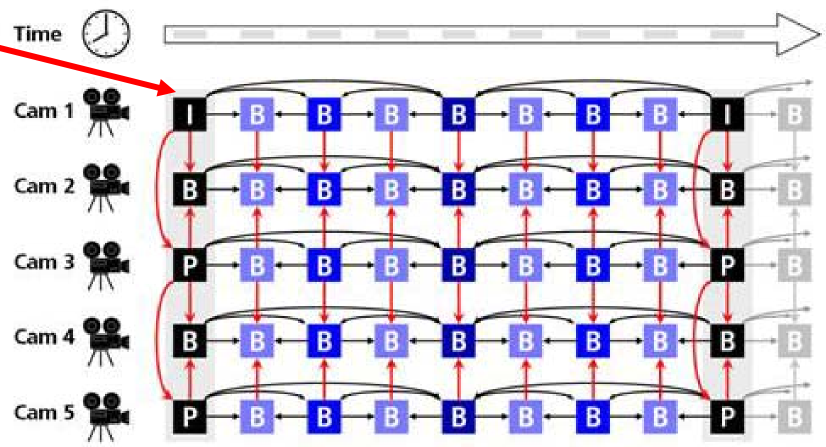
e.g., if each camera image was sent individually as a unique video stream



[Ohm2009]

2) High Profile / FTV Multiview Video

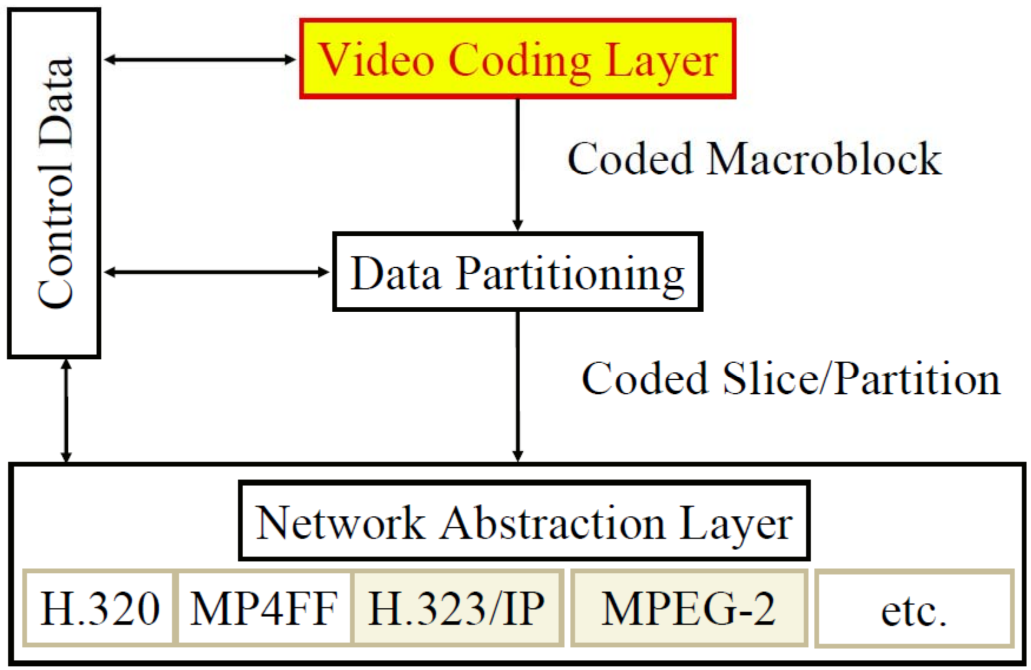
- Multiview video contains a large amount of inter-view statistical dependencies, therefore those dependencies can be exploited.



[Smolic2008][OhmSullivan2005]

MVC (Multiview Coding) Layering

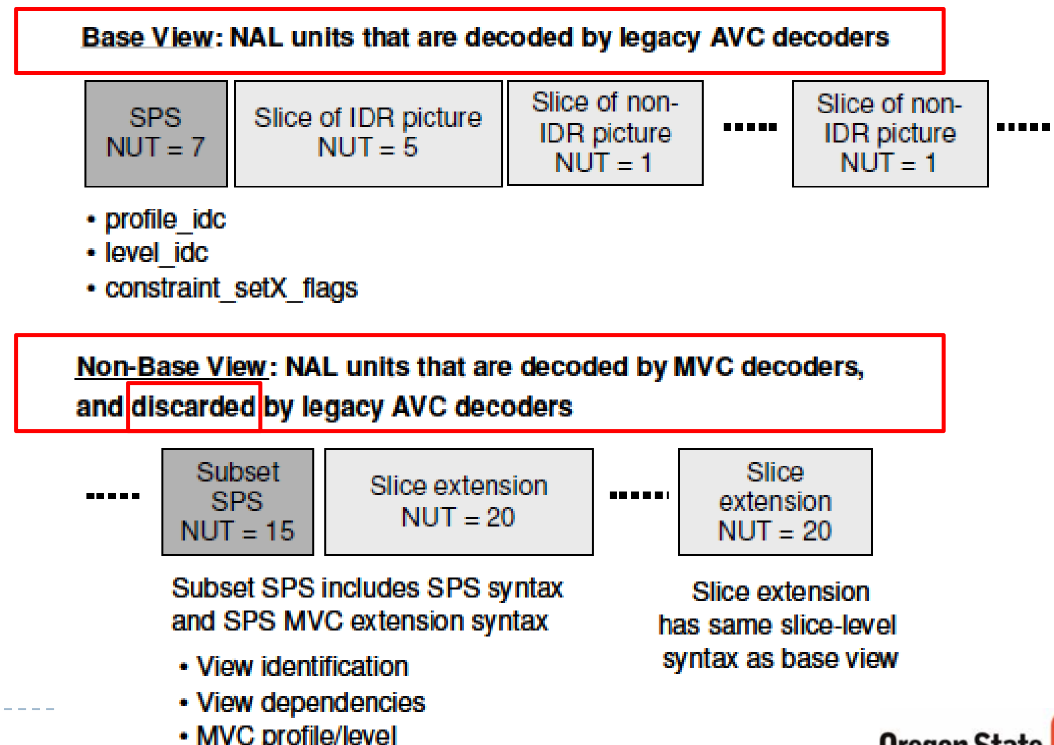
- MPEG-4 SVC (Scalable Video CODEC) allows for the dynamic video quality reception based on differing receiver input bandwidths across an entire system.
- The base layer is always used. The image information is encoded at the Video Coding Layer (VCL) and Transported in the higher Network Abstraction Layer (NAL).



[Polycom2010][UittoVehkapera2013] [Rimac-DjljeNemčićVranješ2008] [WiegandSullivan2003]

MVC (Multiview Coding) Layering

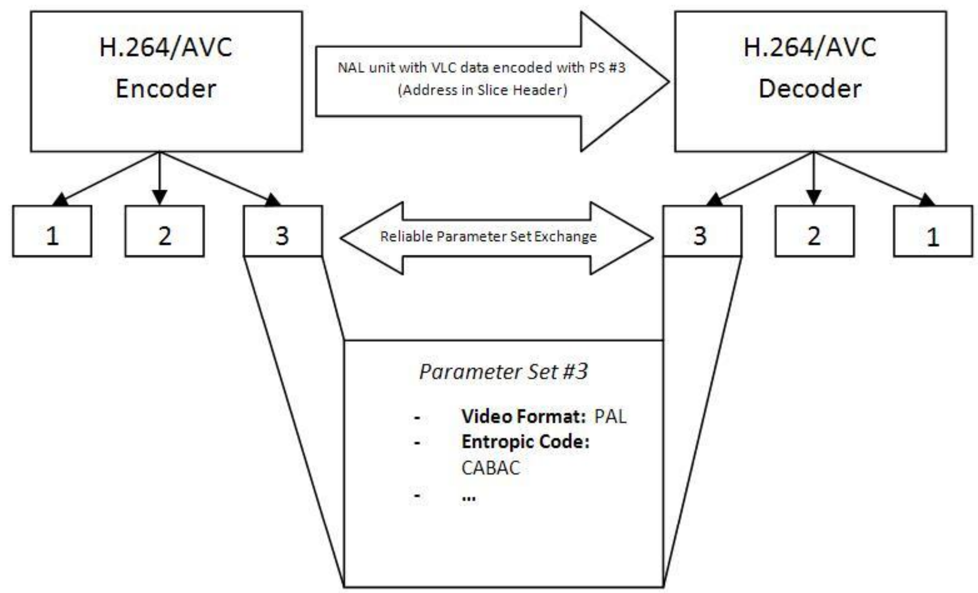
- NAL messages are call "units"
- There are <u>multiple</u> "types" of NAL units that convey both VCL and non-VCL information.
- Each NAL unit type is called an NAL Unit Type ("NUT").



[SchierlNarasimhan2011]

MVC (Multiview Coding) Layering

- MVC exploits significant sharing of common information between views.
- Common (non-VCL) information for all views can be sent via a separate communications path than the VCL data
  - SEI (Supplemental Enhancement Information)
  - Parameter Sets



Agenda

- Background
  - Video Compression and SVC (Scalable Video Coding)
- Multiview Video Types
  - Multiview Coding (MVC) Types and Industry Standards
- State of the Industry – FTV (Free Viewpoint TV)
- OLFVmv (Optimized Live Free Viewpoint multiview video)
  - Motivation
  - Contribution
  - Architecture
  - Algorithms
- Simulation Results
- Extensions to OLFVmv Using Network Coding
- Further Optimization of OLFVmv – Ph.D. Thesis Work
- Conclusion

State of the Industry for FTV MVC

"Today, there are no known streaming services that provide MVV [Multi-View Video] content to home users … the fundamental reasons for this can be listed as:

- *(i) lack of specifications for MVV, such as resolution and number of views, making it difficult to create universal content that is suitable for all multiview displays;*
- *(ii) heterogeneous bandwidth requirement of different multiview displays, making it infeasible to perform transmission over fixed bit-rate channels …"*

 [Dufaux2013] at pg. 201.

State of the Industry for FTV MVC

The last apparent effort to drive standardization related to FTV *transport* appears to be a European initiative called "DIOMEDES" (DIstribution Of Multi-view Entertainment using content aware DElivery Systems).

… which primarily ended in 2012

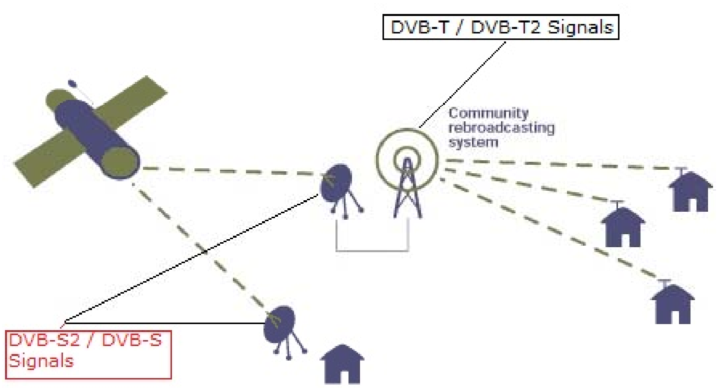
… and was never deployed

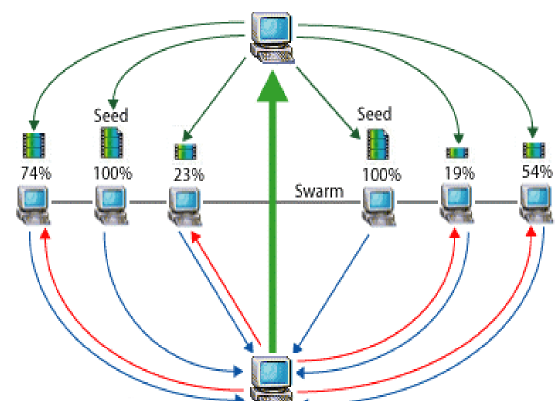
[DIOMEDES D2.3 2012][DIOMEDES D3.6 2011][DIOMEDES D4.4 2011]
[DIOMEDES D4.5 2011][DIOMEDES D2.3 2012]

DIOMEDES – A Closer Look

DIOMEDES offered:

- **<u>A DVB (Direct Video Broadcast)</u>** / DTH (Direct to Home) medium
  - Digital Video Broadcasting (DVB) is a set of standards that define digital broadcasting using existing satellite, cable, and terrestrial infrastructures
- **<u>Combined with a P2P (Peer to Peer) medium</u>**
  - A P2P network is created when two or more PCs are connected and share resources without going through a separate server computer





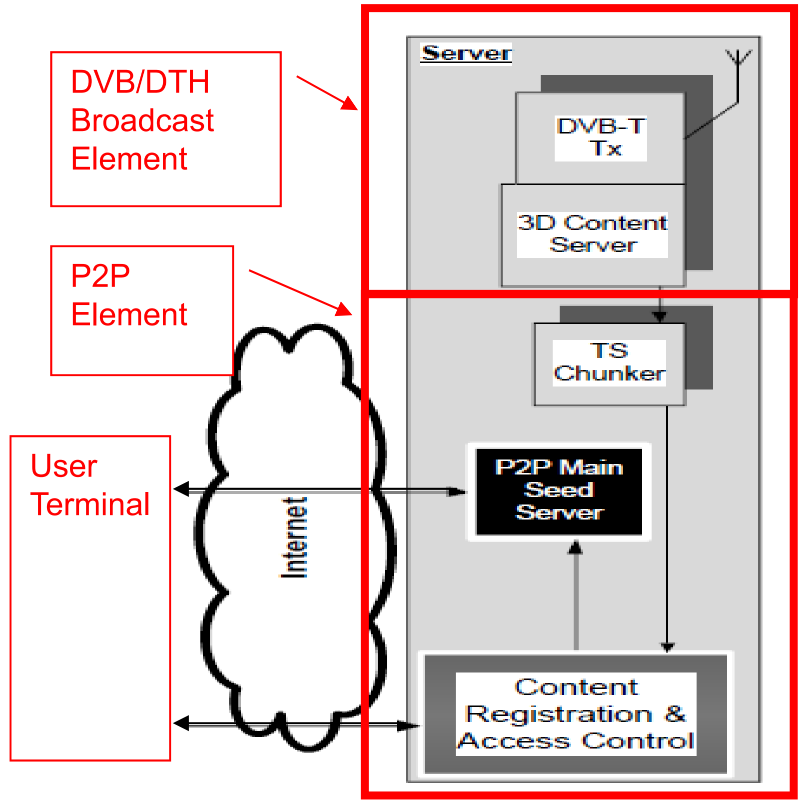
[http://computer.howstuffworks.com/bittorrent2.htm]

[http://www.trackdish.com/wp-content/uploads/2014/01/terrestrial-satellite.jpg]

DIOMEDES – A Closer Look

- DIOMEDES offered a DVB (Direct Video Broadcast) / DTH (Direct to Home) medium combined with a P2P (Peer to Peer) medium

Notably in DIOMEDES, the P2P Main Seed Server did not provide a feedback mechanism related to the desired content from the user terminals.



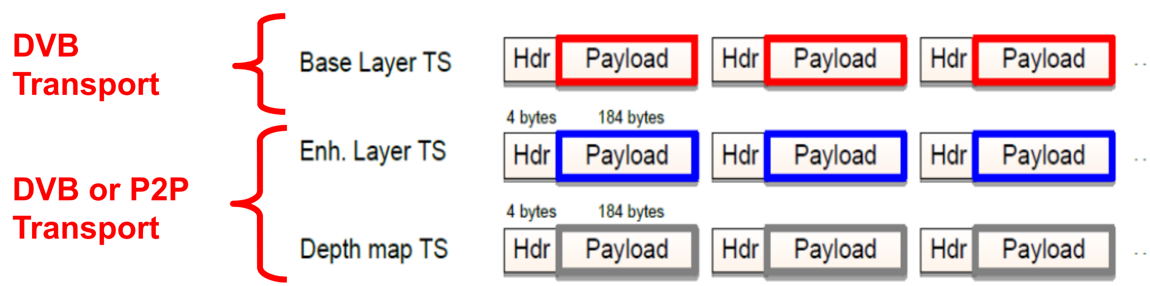
DIOMEDES DVB + P2P FTV Architecture

[DIOMEDES D2.3 2012]

DIOMEDES – A Closer Look

- DIOMEDES employed SVC layering for the:
  - Base layer
  - Metadata layer (e.g., depth map)
  - Enhanced Layer

Based on the layer, either the DVB or P2P medium was used

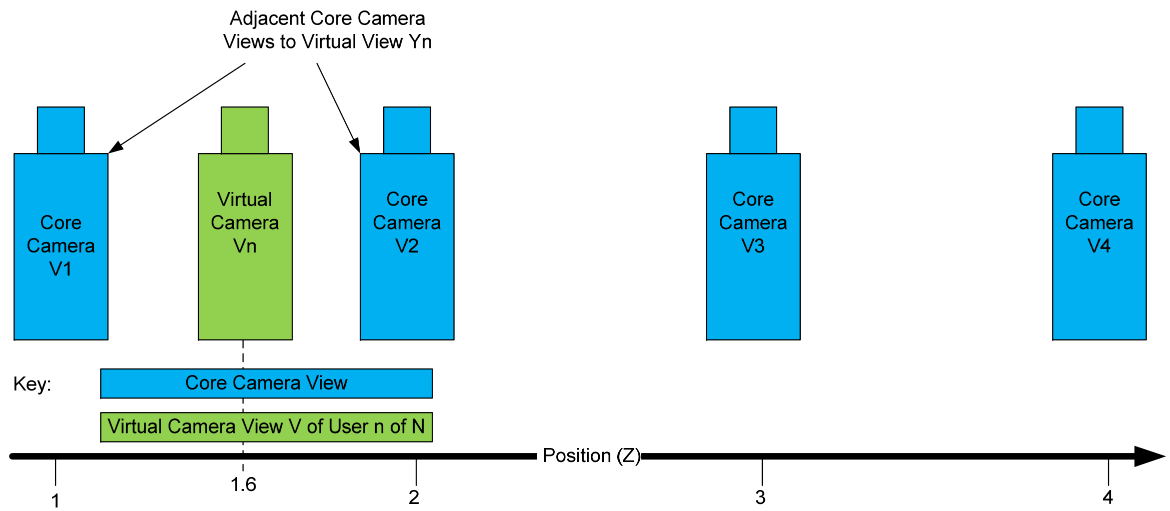


Splitting of content into multiple Transport Streams

[DIOMEDES D4.5 2011]

DIOMEDES – A Closer Look

- Using an array of "Core Cameras" (e.g.,V1,V2, …) virtual views are created using two adjacent core camera views



DIOMEDES – A Closer Look

- In DIOMEDES, video content GOP (Group of Picture) "Chunks" were assigned priorities 1-16, where the Base and Metadata layers shared the same priority for core cameras V2-V8

Chunk priorities P1-7 assigned to the three camera group (V1-3)

View priority order (View-ID)	V1	V2	V3	V4	V5	V6	V7	V8
Base PID	P=1	P=3	P=4	P=8	P=10	P=12	P=14	P=16
Enhancement PID	P=5	P=6	P=7	P=9	P=11	P=13	P=15	P=17
Depth PID	P=2	P=3	P=4	P=8	P=10	P=12	P=14	P=16

Prioritization of GOP chunks over transport streams

DIOMEDES take-aways:

- There is no feedback mechanism from the user terminals on what video is most desired

… but rather, the video content priority is based on core camera "V1"

- DIOMEDES requires impractical broadcast channel bandwidth

[DIOMEDES D3.6 2011]

Agenda

- Background
  - Video Compression and SVC (Scalable Video Coding)
- Multiview Video Types
  - Multiview Coding (MVC) Types and Industry Standards
- State of the Industry – FTV (Free Viewpoint TV)
- OLFVmv (Optimized Live Free Viewpoint multiview video)
  - Motivation
  - Contribution
  - Architecture
  - Algorithms
- Simulation Results
- Extensions to OLFVmv Using Network Coding
- Further Optimization of OLFVmv – Ph.D. Thesis Work
- Conclusion

Motivation – for Optimized Live Free Viewpoint multiview video (OLFVmv)

1) There is a need for ***a practical transport*** of FTV over ***existing broadcast*** mediums:
   - **Using DVB** bandwidths more efficiently

2) There is a need for ***a practical transport*** of FTV over P2P networks:
   - **Enable low bandwidth, mobile P2P** networks using as little as 2 Mbps

3) Offering **superior performance**

[ETSI EN300 429 V1.2.1:1998][ETSI EN200 421 V1.1.2:1997][ETSI EN302 307:2009]
[ETSI EN302 304 V1.1.1:2004][ETSI TS102 034 V1.4.1:2009][ETSI EN 302 755 V1.2.1:2010]

Motivation

Prioritize Content *based on what people want to see…*

... because not all content is equally important to the overall viewing audience.

Core Camera | Core Camera | Core Camera | Core Camera

Contribution – Optimized Live Free Viewpoint multiview video (OLFVmv)

1) Improvements / Contributions:

- (a) An improved architecture
- (b) Intelligent algorithms combined with the improved architecture ***to predict what video content is important***

2) A roadmap to other improvements:

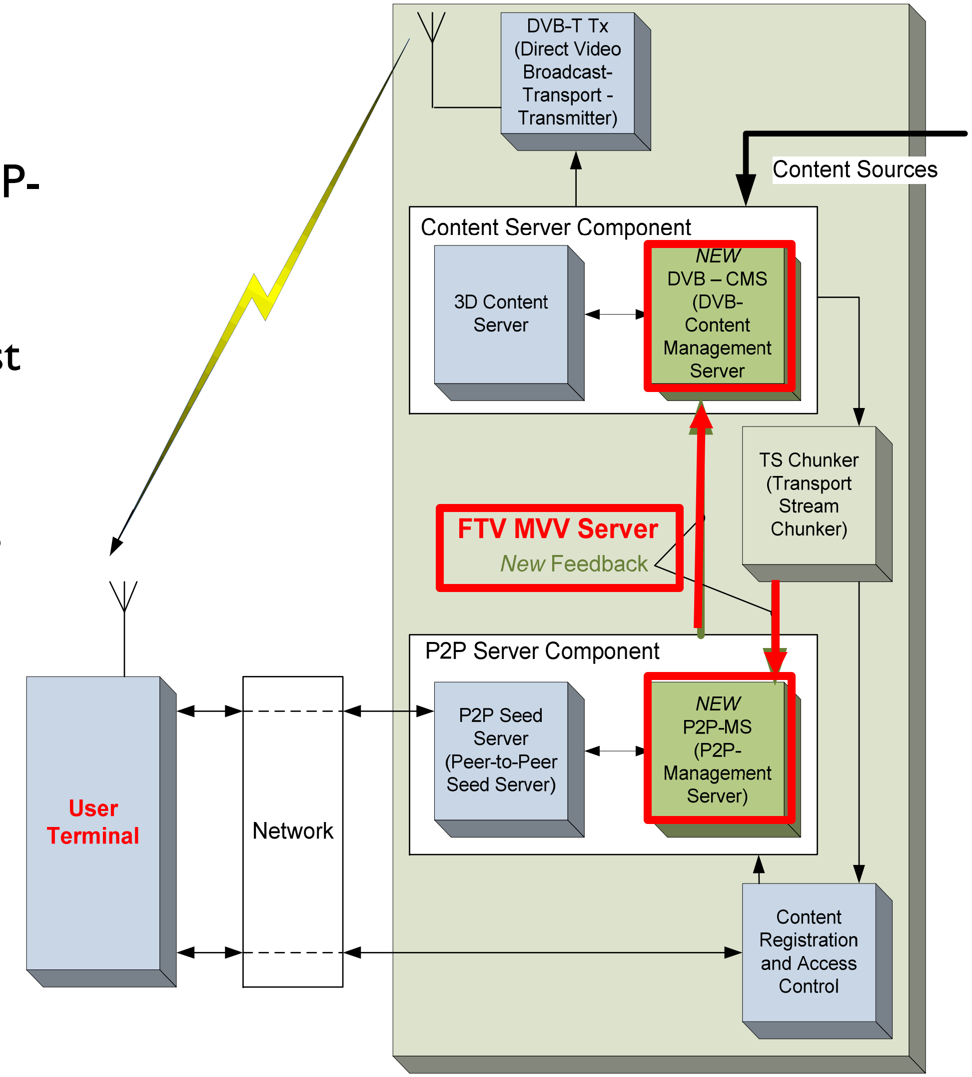
- Network coding for the remaining data to be sent via the P2P network

OLFVmv – Improved System Architecture

Improvements:

- A new "P2P-Management Server" ("P2P-MS") to perform the content selection/prioritization algorithms and provide feedback to the DVB broadcast system, and

- A "DVB-Content Management Server" ("DVB-CMS") to receive input from the P2P-MS on the most prevalent (requested views) video content to broadcast, and based on that input, to select the most prevalent content to be broadcast over the two DVB broadcast channels



Oregon State UNIVERSITY OSU

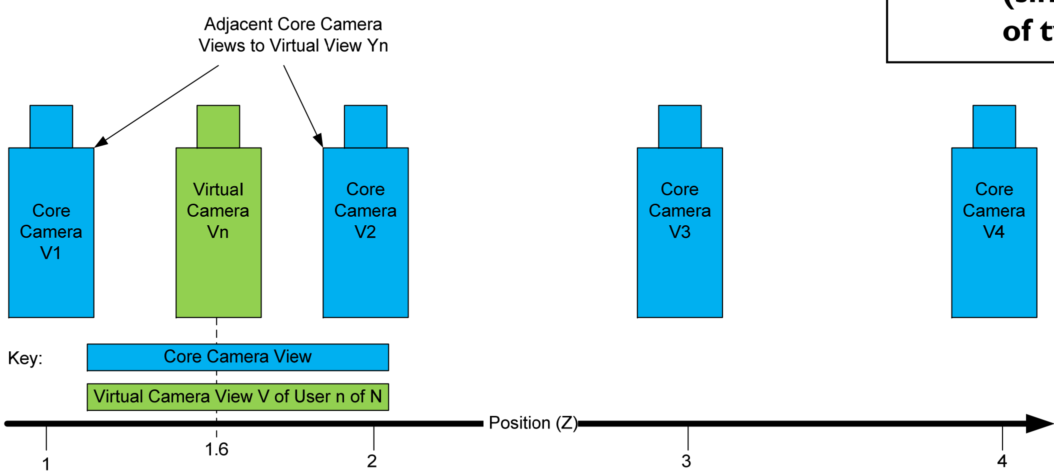
OLFVmv – Algorithms Overview

- The OLFVmv system utilizes each viewers desired viewing position, "Vn"

For discussion purposes:

Integer "Vx" is defined to be a specific primary core camera view for a user, called the *left* core camera view, and "Vx+1" is the core camera view immediately adjacent to the *right* of "Vx".

Non-Integer "Vn" is defined as a synthetic (simulated) desired view from the combination of two adjacent core camera views.



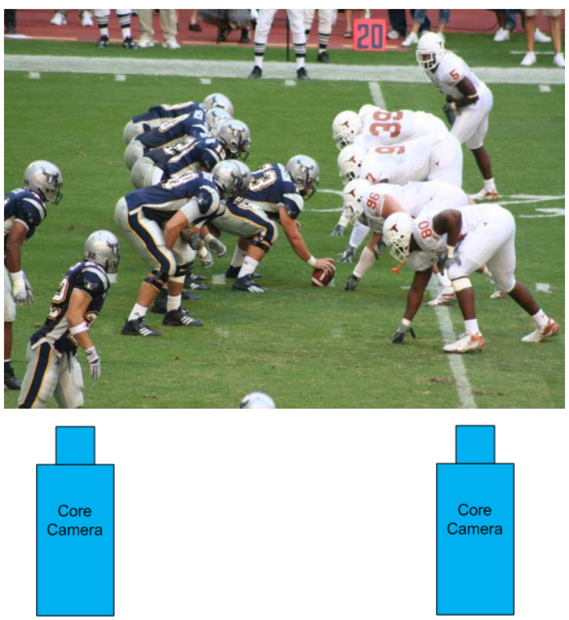
Core camera registration based on virtual camera view Vn.

OLFVmv – Algorithms Overview

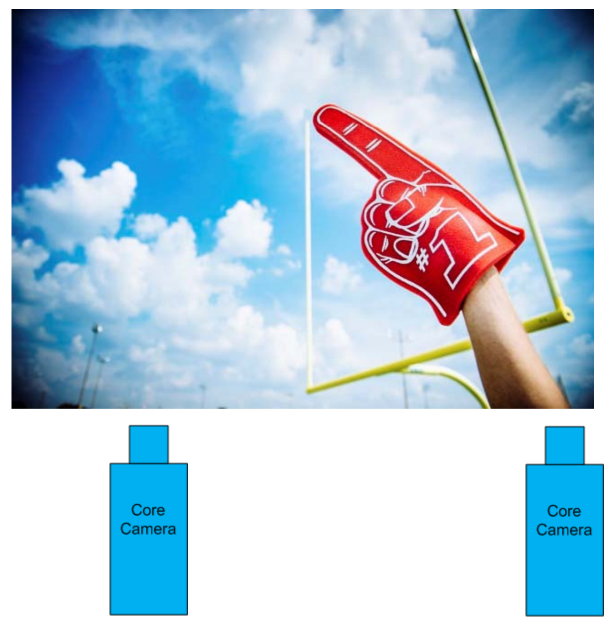
With limited resources over the DVB and P2P networks, determine: what is content is most relevant?

How? By developing intelligent algorithms

How many viewers want to see this **and what is the trend**?



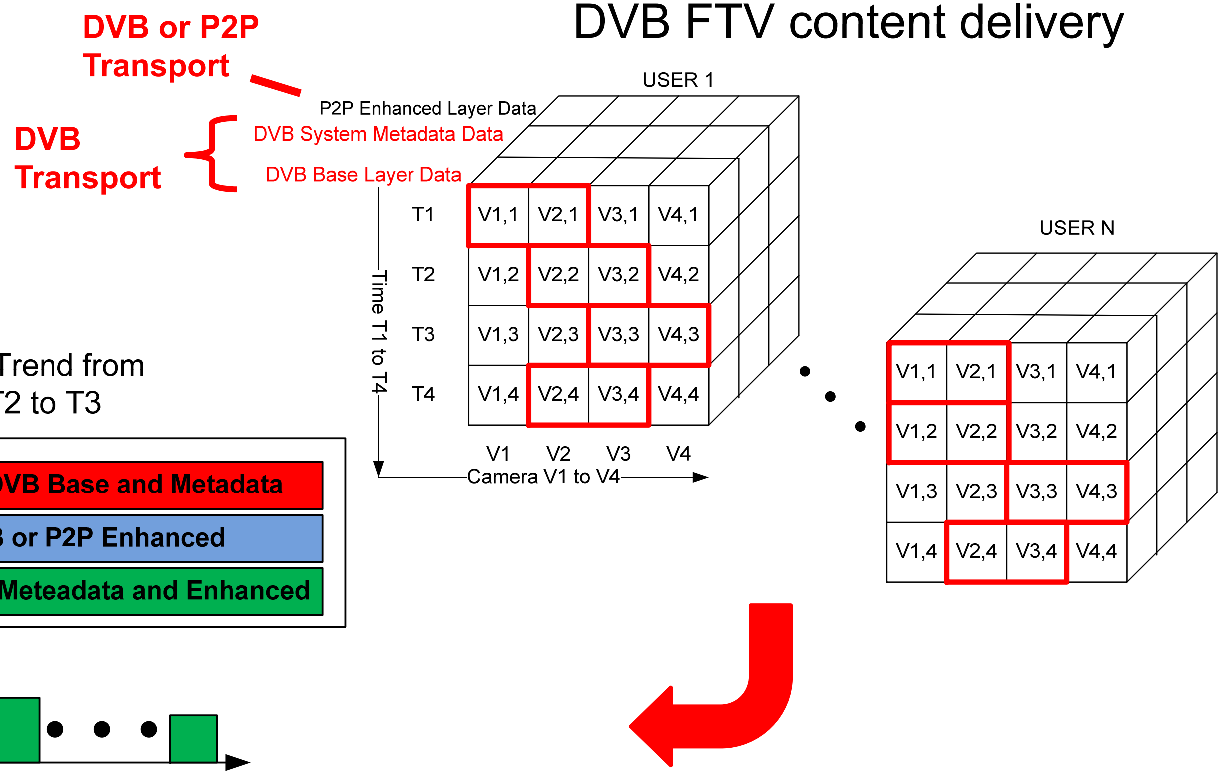
How many viewers want to see this **and what is the trend**?

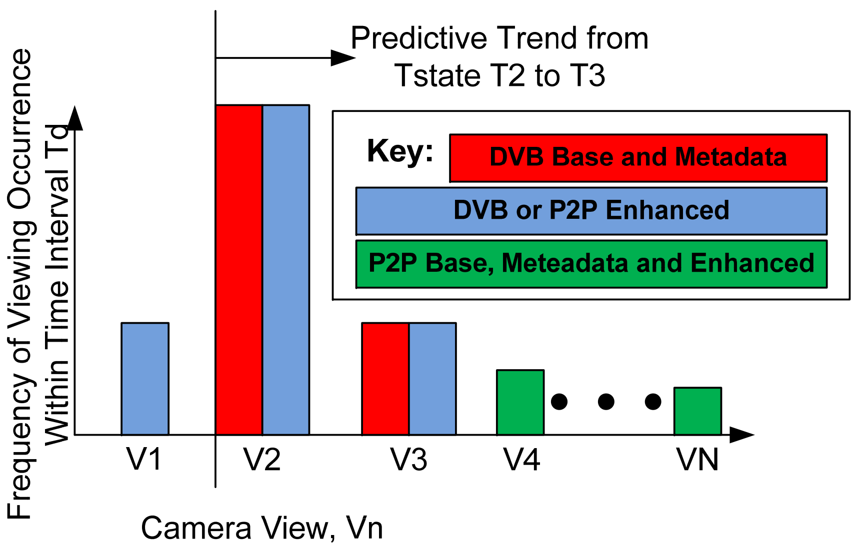


OLFVmv – Algorithms Overview

- Track viewing patterns over time
- Predict trends

DVB FTV content delivery





OLFVmv viewing trend modeling

OLFVmv – Algorithms
Step 1 – Assign Core Camera Views

Step 1: Let a system of N=10 viewer have the following viewing pattern…

See Appendix C for a complete list and explanation of variables.

```
%% Camera view registration algorithm

function register_cameras_views

global Num_Cameras
global Vn_Viewer_View
global Viewer_Time_Osc_Position
global Viewer_Random_Position_Offset

% Virtual desired view = mean viewing position, plus/minus random offset
% This operation takes the offset matrix [Num_Viewers x 1] and adds to the
% View Position matrix [1 x Time Ticks] and equals a view position for each
% viewer = [Num_Viewers x Time Ticks] matrix

Vn_Viewer_View.Vn ...
   = min(max((Viewer_Time_Osc_Position + Viewer_Random_Position_Offset),1),Num_Cameras);

% Calculate left most camera by truncating Vn in to an integer with the
% minimum camera being Camera = 1 and the right most camera being one from
% the, Camera = Num_Cameras - 1
Vn_Viewer_View.Vn_Left = floor(min(max(Vn_Viewer_View.Vn,1),Num_Cameras - 1));

% Calculate right most camera by taking Vn_Viewer_View.Vn_Left and adding
% one.  The case should never exist where the Right most camera exceeds
% Num_Cameras, but if it does, limit it to Num_Cameras
Vn_Viewer_View.Vn_Right = int8(min((Vn_Viewer_View.Vn_Left + 1),Num_Cameras));

fprintf('execution complete: register_cameras_views \n')

end
```

Algorithm

User (N)	Vn Current View
1	1.6
2	1.2
3	1.5
4	2.1
5	1.7
6	2.0
7	1.7
8	1.3
9	4.0
10	2.0

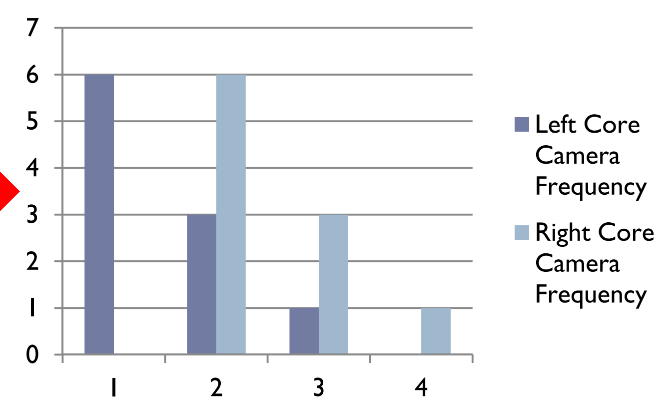
OLFVmv – Algorithms
Step 2 - Histogram Creation

Step 2: Use previous output to create histogram

Continued from Figure 17, above.

```
%% Algorithm to find left and right core camera views to transmit over the DVB
medium based on building a histogram

function build_viewing_histogram

global Vn_Viewer_View
global Channel_Histogram
global Num_Sim_Run_Time_Ticks
global Num_Cameras

% define bins 0.5-1.5, 1.5-2.5, and so on to capture the center of each bin
% at 1, 2, 3, 4, ... Num_Cameras, camera views into Num_Cameras discrete bins
bin_edges = 0.5:1:Num_Cameras+0.5;

% for each time tick, do a histogram of camera number (Vn) viewed, by the
% total population of all viewers

for Histogram_Time_Index = 1:Num_Sim_Run_Time_Ticks
    Channel_Histogram(:,Histogram_Time_Index)...
        = histcounts(Vn_Viewer_View.Vn_Left(:,Histogram_Time_Index),bin_edges);
end % end - for

fprintf('execution complete: build_viewing_histogram \n')

end % end - build_viewing_histogram
```

User (N)	Vn Current View	Vn_Left[N] Left Core Camera	Vn_Right[N] Right Core Camera
1	1.6	1	2
2	1.2	1	2
3	1.5	1	2
4	2.1	2	3
5	1.7	1	2
6	2.0	2	3
7	1.7	1	2
8	1.3	1	2
9	4.0	3	4
10	2.0	2	3



Oregon State University OSU

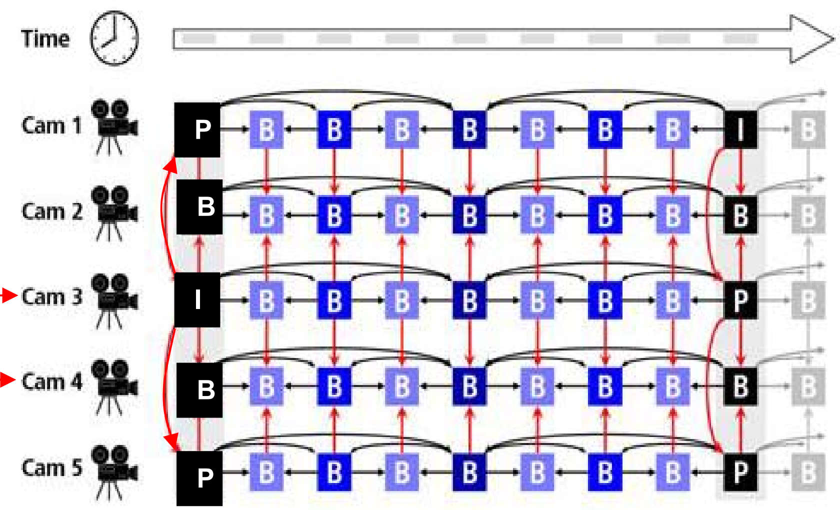
OLFVmv – Algorithms
Step 3 – Determine DVB Channel Content

Step 3: Determine most prevalent LEFT and RIGHT camera views and assign those to the DVB channels

Example: Let LEFT camera = V3, and RIGHT camera be V4

Left core camera view:

Right core camera view: Uses only P-Frames and/or B-Frames derived from the left core camera view



Temporal P and B-Frame view prediction structure for MVC

Core Camera View →	V1	V2	V3	V4	V5	V6	V7	V8
Base Layer	P2P	P2P	DVB	DVB	P2P	P2P	P2P	P2P
Metadata Layer	P2P	P2P	DVB	DVB	P2P	P2P	P2P	P2P
Enhanced Layer	P2P	P2P	DVB	DVB	P2P	P2P	P2P	P2P

Example of video content distribution between the DVB and P2P channels

V_{DVB} = BLUE

V_{P2P} = GREEN

OLFVmv – Algorithms
Step 4 – Determine P2P Channel Content

Step 4: Set the LEFT and RIGHT most prevalent base, metadata and enhanced layers to a priority such that the content is transported over the DVB channel

… and everything else over the P2P channel

Let $\mathbf{V_{ALL}} = V_{Meta}[m] + V_{Base}[m] + V_{Enhanced}[m])$

Then:

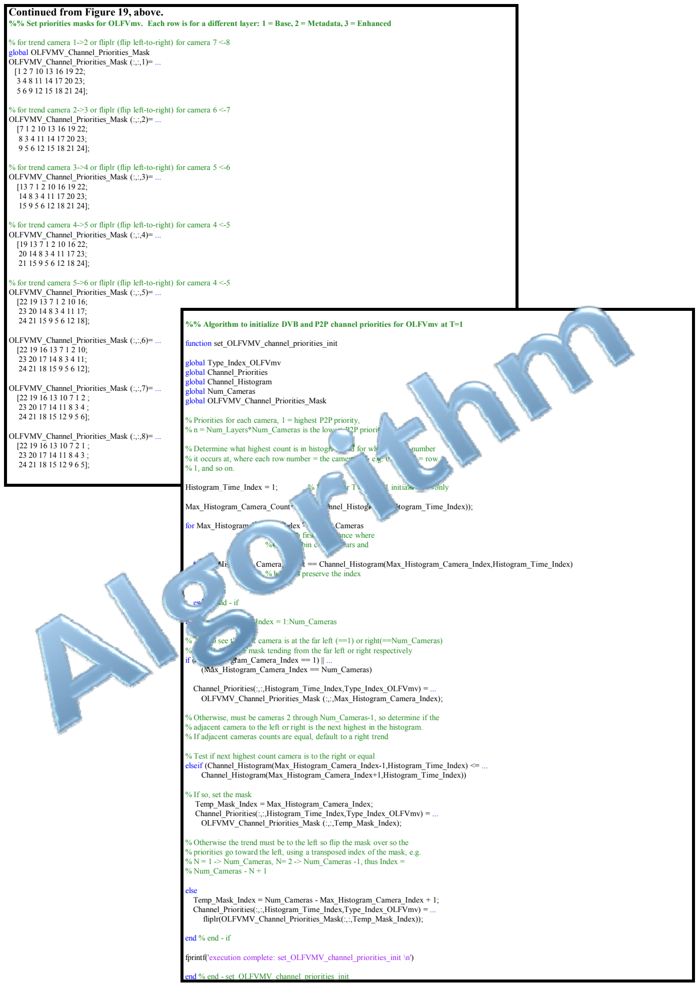
$$V_{P2P} = \mathbf{V_{ALL}} - V_{DVB}$$

OLFVmv – Algorithms
Step 4 – Set DVB and P2P Content Priorities

V_{DVB} = BLUE	
V_{P2P} = GREEN	

Priority:
1= Most Important
24 = Least Important

Core Camera View →	V1	V2	V3	V4	V5	V6	V7	V8
Base Layer	P2P	P2P	DVB	DVB	P2P	P2P	P2P	P2P
Metadata Layer	P2P	P2P	DVB	DVB	P2P	P2P	P2P	P2P
Enhanced Layer	P2P	P2P	DVB	DVB	P2P	P2P	P2P	P2P



Algorithm Input

OLFVmv video content distribution between the DVB and P2P channels at T-State = 1

Core Camera View →	V1	V2	V3	V4	V5	V6	V7	V8
Base Layer	13	7	1	2	10	16	19	22
Metadata Layer	14	8	3	4	11	17	20	23
Enhanced Layer	15	9	5	6	12	18	21	24

OLFVmv – Algorithms
Step 5 – Establish a Trend/Update Priorities

Continued from Figure 25, above.

```
%% Figure 27 from Thesis for  - Set DVB and P2P channel priorities for OLFVmv

function set_OLFVMV_channel_priorities

global Channel_Priorities
global Channel_Histogram
global Num_Cameras
global OLFVMV_Channel_Priorities_Mask
global Num_Sim_Run_Time_Ticks
global Type_Index_OLFVmv

% Priorities for each camera, lowest number = highest P2P priority,
% highes number n = Num_Layers*Num_Cameras is the lowest P2P priority

% Starting with T=2 and for each time tick after, take the histogram results
% from the last T-State, compare them to the current T-State, determine if the
% trend is to the left or right and set the priority mask accordingly

Max_Histogram_Camera_Index_Last = 1;          % Initialized the previous T-State as 1

for Histogram_Time_Index = 2:Num_Sim_Run_Time_Ticks

   % Determine what highest count is in histogram for this T-State and
   % what camera number each occurred at.  The function find returns row
   % and column so this needs to be reduced to just column.

   Max_Histogram_Camera_Count_Current =
max(Channel_Histogram(:,Histogram_Time_Inde

   for Max_Histogram_Camera_Index      rren      N      _Cameras
                                  %   ind the firs      urre
                                        x bin c     occu

      if Max_Histogram_      era            urrent == ...
Channel_Histogram(Max      ogram_Camera_Index_Current,Histogram_Time_Index)
                                  %   break to preserve the index
         break

      end % end - if

   end % end - for Max_Histogram_Camera_Index_Current = 1:Num_Cameras

   % Test to see the max camera is at either end, and if so, set the mask
   % trending from the end

   if (Max_Histogram_Camera_Index_Current == 1)||...
        (Max_Histogram_Camera_Index_Current == Num_Cameras)

Channel_Priorities(:,:,Histogram_Time_Index,Type_Index_OLFVmv) = ...

OLFVMV_Channel_Priorities_Mask(:,:,Max_Histogram_Camera_Index_Current);

% Otherwise, must be cameras 2 through Num_Cameras-1, so determine if the
   % trend is from the left to right.  Default is to the right.

   % Compare where the current max camera histogram point is the current
   % T-State compared to where it was in the last T-State.  If the current
   % T-State index is greater, then the trend is to

   elseif (Max_Histogram_Camera_Inde        >=
Max_Histogram_Camera_Index_La

    % If so, set the ma
      Temp_M  sk_Ind      lax_      gra   _Camera_Index_Current;

Char    _Prio     :,:,Histo      _Time_Index,Type_Index_OLFVmv) = ...
        O     MV_      _Priorities_Mask(:,:,Temp_Mask_Index);

    therwise     trend must be to the left so flip the mask over so the
     orities go toward the left, using a transposed index of the mask, e.g
% N = 1 -> Num_Cameras, N= 2 -> Num_Cameras -1, thus Index =
% Num_Cameras - N + 1

   else
      Temp_Mask_Index = Num_Cameras -
Max_Histogram_Camera_Index_Current + 1;

Channel_Priorities(:,:,Histogram_Time_Index,Type_Index_OLFVmv) = ...
        fliplr(OLFVMV_Channel_Priorities_Mask
(:,:,Temp_Mask_Index));

   end % end - if

% now that we are done testing, remember the index where the max camera
% occurred for the next loop.  Thus _Current becomes _Last.

Max_Histogram_Camera_Index_Last =
Max_Histogram_Camera_Index_Current;

end % end - for Histogram_Time_Index = 2:Num_Sim_Run_Time_Ticks

fprintf('execution complete: set_OLFVMV_channel_priorities \n');

end % end - function set_OLFVMV_channel_priorities
```

V_{DVB} = BLUE	
V_{P2P} = GREEN	

OLFVmv video content distribution between the DVB and P2P channels at T-State = 1

Core Camera View →	V1	V2	V3	V4	V5	V6	V7	V8
Base Layer	13	7	1	2	10	16	19	22
Metadata Layer	14	8	3	4	11	17	20	23
Enhanced Layer	15	9	5	6	12	18	21	24

Histogram Update / Determine Trend

Viewing trend from T-State =1 to T-State = 2 (Left core view)

Viewing trend from T-State =1 to T-State = 2 (Right core view)

Core Camera View →	V1	V2	V3	V4	V5	V6	V7	V8
Base Layer	19	13	7	1	2	10	16	22
Metadata Layer	20	14	8	3	4	11	17	23
Enhanced Layer	21	15	9	5	6	12	18	24

OLFVmv video content distribution between the DVB and P2P channels at T-State = 2…N

DIOMEDES Versus OLFVmv Priorities

In contrast, DIOMEDES transports 3 **fixed** channels (V1, V2, and V3) via DVB

For the P2P channels, DIOMEDES sets the Base and Metadata layers at a higher priority than the Enhanced layer

Core Camera View →	V1	V2	V3	V4	V5	V6	V7	V8
Base Layer	1	3	4	8	10	12	14	16
Metadata Layer	2	3	4	8	10	12	14	16
Enhanced Layer	5	6	7	9	11	13	15	17

V_{DVB} = BLUE
V_{P2P} = GREEN

[DIOMEDES D3.6 2011]

Agenda

- Background
  - Video Compression and SVC (Scalable Video Coding)
- Multiview Video Types
  - Multiview Coding (MVC) Types and Industry Standards
- State of the Industry – FTV (Free Viewpoint TV)
- OLFVmv (Optimized Live Free Viewpoint multiview video)
  - Motivation
  - Contribution
  - Architecture
  - Algorithms
- **Simulation Results**
- Extensions to OLFVmv Using Network Coding
- Further Optimization of OLFVmv – Ph.D. Thesis Work
- Conclusion

Simulation Results - Baseline Assumptions

Video Content Bandwidth Requirements and Assumptions

Description	***Bandwidth Used per Channel (Mbits/s)***
Video Content Bandwidth Requirements	**For the primary DVB channel (e.g., left most DVB channel)** • Primary Base (B) Layer = 6 Mbps • Primary Metadata (M) Layer = 2 Mbps • Primary Enhanced (E) Layer = 13.3 Mbps → 12 Mbps **Total Primary B+E+M ~ 21.2 Mbps → 20 Mbps** For an adjacent channel (e.g., any P2P or DVB channel other than the primary channel) • Adjacent Base (B) Layer = 6 x (0.25x9+0.5x1)/10 = 1.65 Mbps → 2 Mbps • Adjacent Metadata (M) Layer = 2 Mbps • Adjacent Enhanced (E) Layer = 13.3 x (0.25x9+0.5x1)/10 = 3.66 Mbps → 4 Mbps **Total Primary B+E+M ~ 7.3 Mbps → 8 Mbps**
Channel Capacities	
DVB Channel Throughput (assume DVB-S, ATSC)	Assume 28 Mbps (e.g., 1 primary HDTV 3D channel (B+E+M) plus 1 adjacent HDTV 3D channel (B+E+M)
P2P Channel Throughput (assume no P2P or Network coding)	Mean_P2P_BW_Simulation_Rates = {2, 16, 32 and 64} Mbits/s, with Gaussian distribution of Sigma_P2P_BW = 0.1

Simulation Results - Baseline Assumptions

Based on the assumptions provided, the following is an example of the assignment of bandwidth based on a given OLFVmv priority map:

DVB Channel Priority / Bandwidth
P2P Channel Priority / Bandwidth

Core Camera View →	V1	V2	V3	V4	V5	V6	V7	V8
Base Layer	19	13	7	1	2	10	16	22
Metadata Layer	20	14	8	3	4	11	17	23
Enhanced Layer	21	15	9	5	6	12	18	24

Example OLFVmv Priority Matrix

Total DVB bandwidth = 28 Mbps

Core Camera View →	V1	V2	V3	V4	V5	V6	V7	V8
Base Layer	2Mbps	2Mbps	2Mbps	6Mbps	2Mbps	2Mbps	2Mbps	2Mbps
Metadata Layer	2Mbps	2Mbps	2Mbps	2Mbps	2Mbps	2Mbps	2Mbps	2Mbps
Enhanced Layer	4Mbps	4Mbps	4Mbps	12Mbps	4Mbps	4Mbps	4Mbps	4Mbps
Total	8Mbps	8Mbps	8Mbps	20Mbps	8Mbps	8Mbps	8Mbps	8Mbps

Corresponding OLFVmv DVB Transport Bandwidth

Simulation Results - Baseline Assumptions

Based on the assumptions provided, the allocation for DIOMEDES is as follows:

Core Camera View →	V1	V2	V3	V4	V5	V6	V7	V8
Base Layer	1	3	5	10	13	16	19	22
Metadata Layer	2	4	6	11	14	17	20	23
Enhanced Layer	7	8	9	12	15	18	21	24

DIOMEDES Priority Matrix

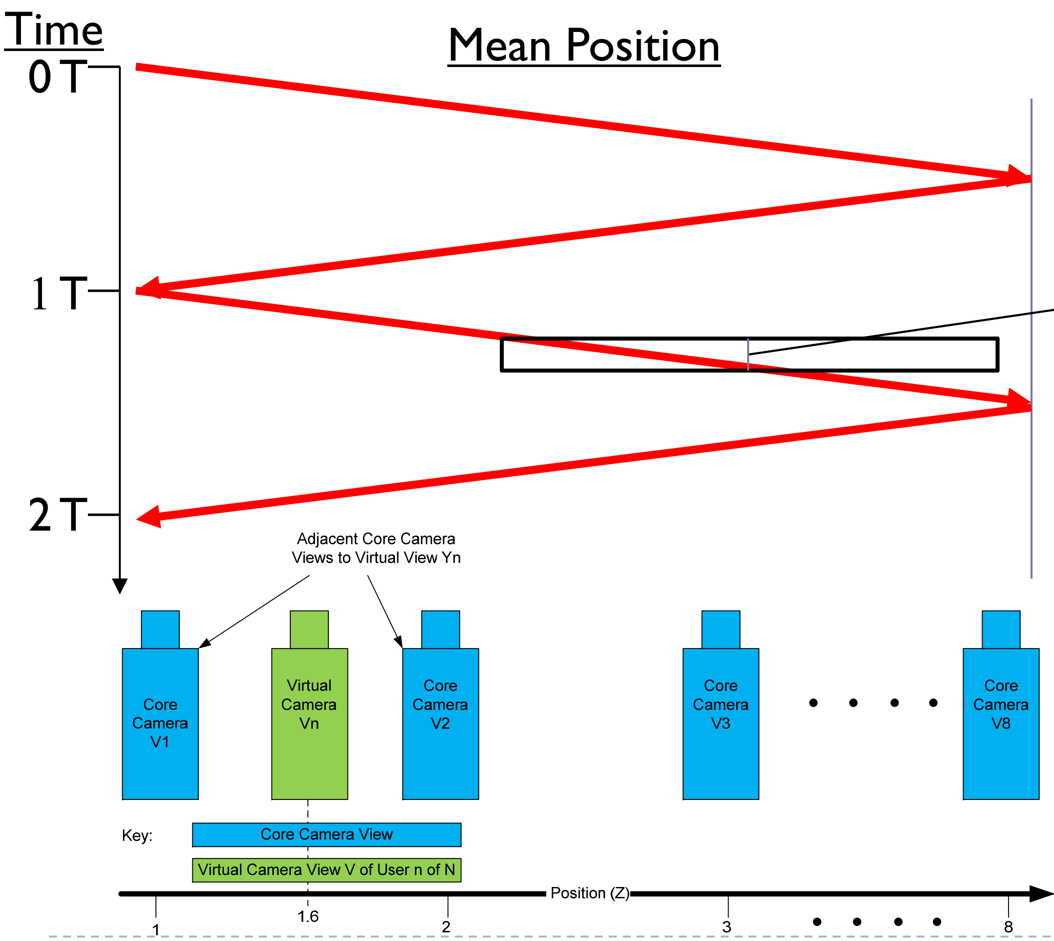
Total DVB bandwidth = 28 Mbps

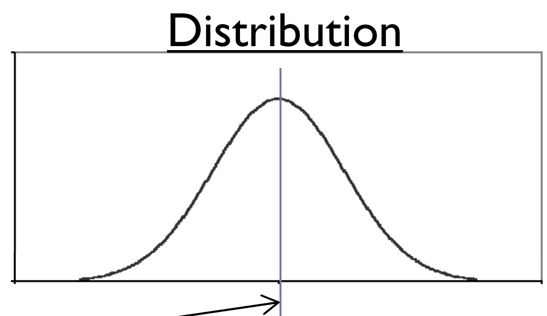
Core Camera View →	V1	V2	V3	V4	V5	V6	V7	V8
Base Layer	6Mbps	2Mbps	2Mbps	2Mbps	2Mbps	2Mbps	2Mbps	2Mbps
Metadata Layer	2Mbps	2Mbps	2Mbps	2Mbps	2Mbps	2Mbps	2Mbps	2Mbps
Enhanced Layer	12Mbps	4Mbps	4Mbps	4Mbps	4Mbps	4Mbps	4Mbps	4Mbps
Total	20Mbps	8Mbps	8Mbps	8Mbps	8Mbps	8bps	8Mbps	8Mbps

Corresponding DIOMEDES DVB Transport Bandwidth

Simulation Parameters – Viewing Position for Each of N=100 Users

Viewer Oscillation Rate Period: T = {5, 10 and 50 seconds} over a field of 8 cameras





Viewer to Viewer Normal Random Distribution about Mean at any give time:
σ scale factor = {0.1, 0.5. 1.0 and 2.0}

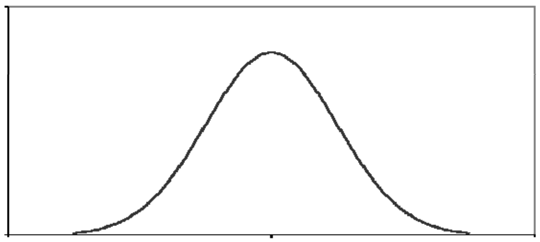
Simulation Parameters – Available P2P Bandwidth for Each of N=100 Viewers

All Simulations run at each of the following mean P2P bandwidths:

Each User's Mean Available P2P Bandwidth	**2**	**16**	**32**	**64**

For each user, at each bandwidth, a normal random variance scale factor of {0.1} x the Mean Bandwidth was applied

For example, assume a viewer with a Std Dev. of -2 and mean bandwidth of 32 Mbps:



-2 x 0.1 x 32 Mbps = -6.4 Mbps

Thus 25.6 Mbps for a given user

Simulation Results

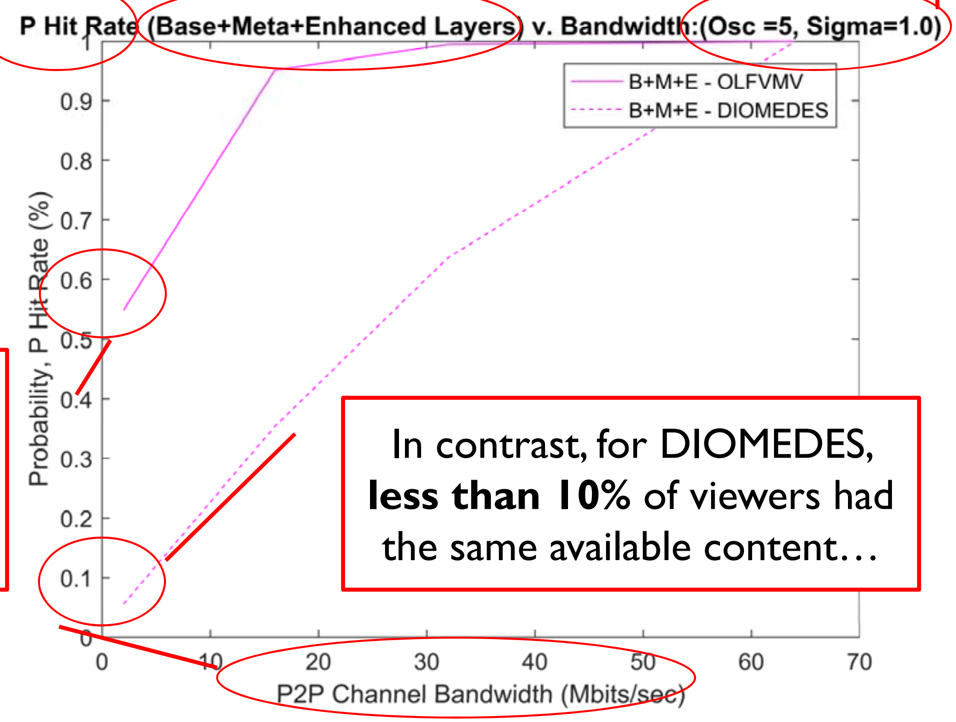
For each of N=100 viewers, the performance of the OLFVmv system was contrasted against DIOMEDES in Matlab

Example output:

The analysis was performed over all viewer viewing patterns with variances

All layer permutations were analyzed

For the given viewing simulation, the "P Hit Rate" represents the probability that the LEFT and RIGHT content was *for (1) a given layer(s) for (2) a given viewer* so that the viewer could synthesized the desired virtual view



In this example for OLFVmv, at 2 Mbps, 58% of the viewers had ALL layers of content to create a synthesized view with a FAST oscillation rate of T=5 sec and sigma = 1.0

In contrast, for DIOMEDES, **less than 10%** of viewers had the same available content…

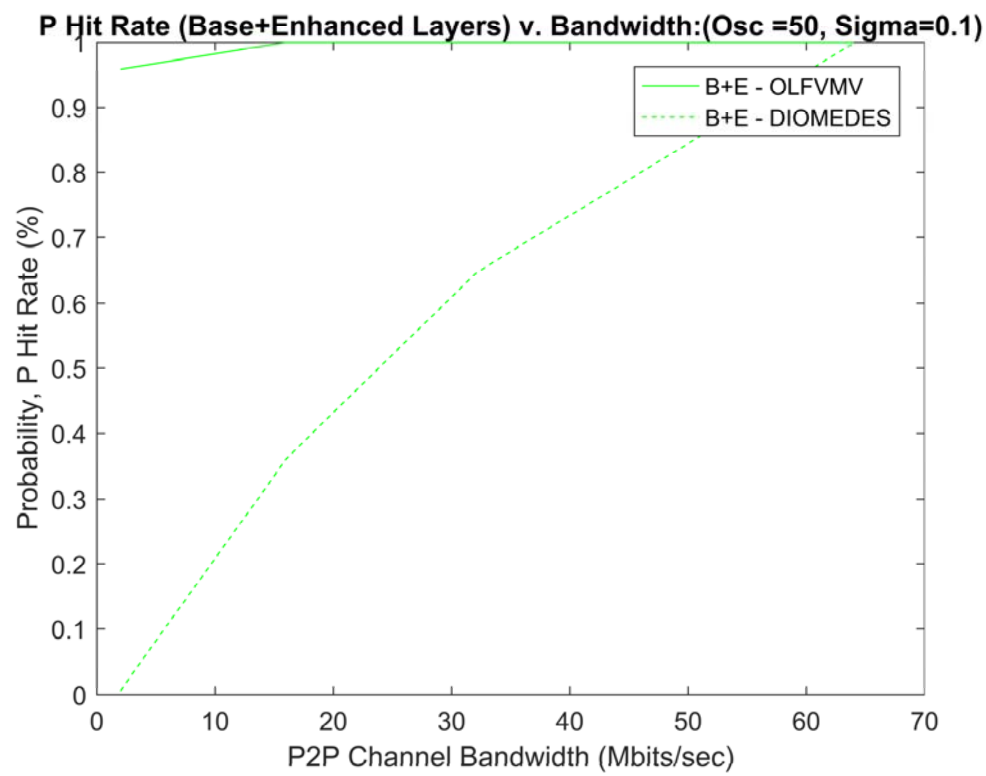
The analysis was performed over all simulated bandwidths including impairments

Oregon State UNIVERSITY OSU

Simulation Results

Overall, OLFVmv outperformed DIOMEDES for P2P bandwidths of less than 64 Mbps

Only at 64 Mbps did DIOMEDES' performance match that of OLFVmv

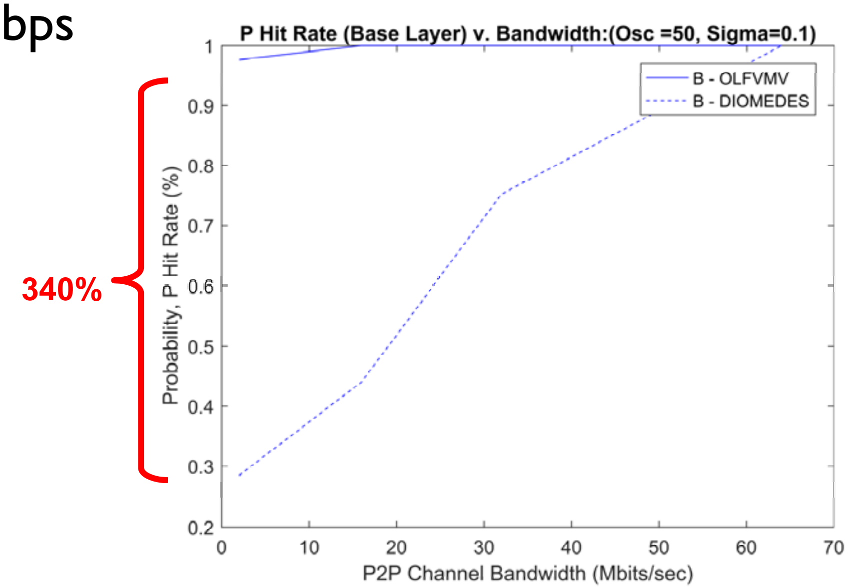


Oregon State University OSU

Simulation Results

At a P2P channel bandwidth = 2 Mbps, a slow viewer oscillation rate (Osc = 50 seconds), and with a small viewer variance (sigma scale factor = 0.1) **OLFVmv outperformed DIOMEDES by 340%.**

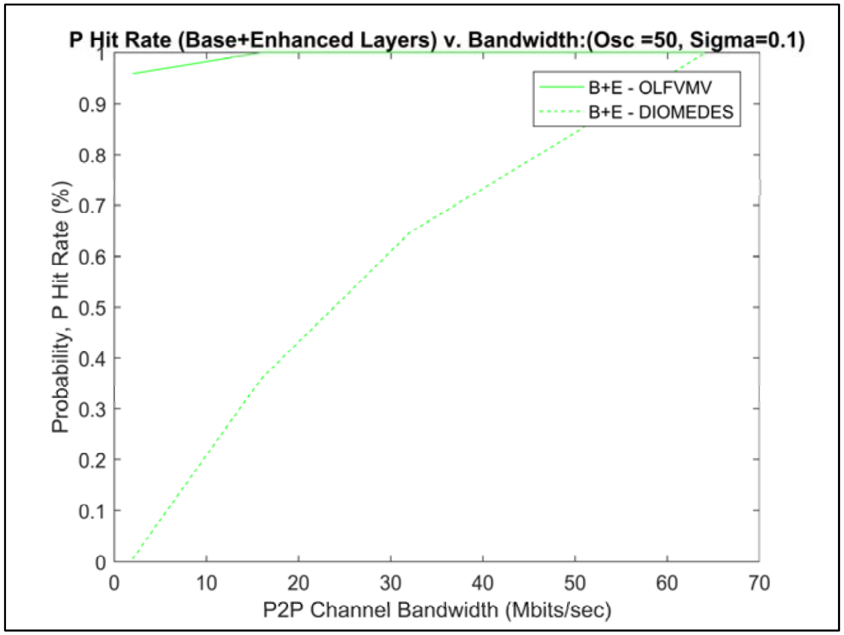
- **Nearly 98%** of OLFVmv viewers had the desired content at **the base layer** available
- … **while only 29% of DIOMEDES** viewers had the desired content **at the base layer -** DIOMEDES was able to obtain parity only at a P2P channel bandwidth of 64 Mbps

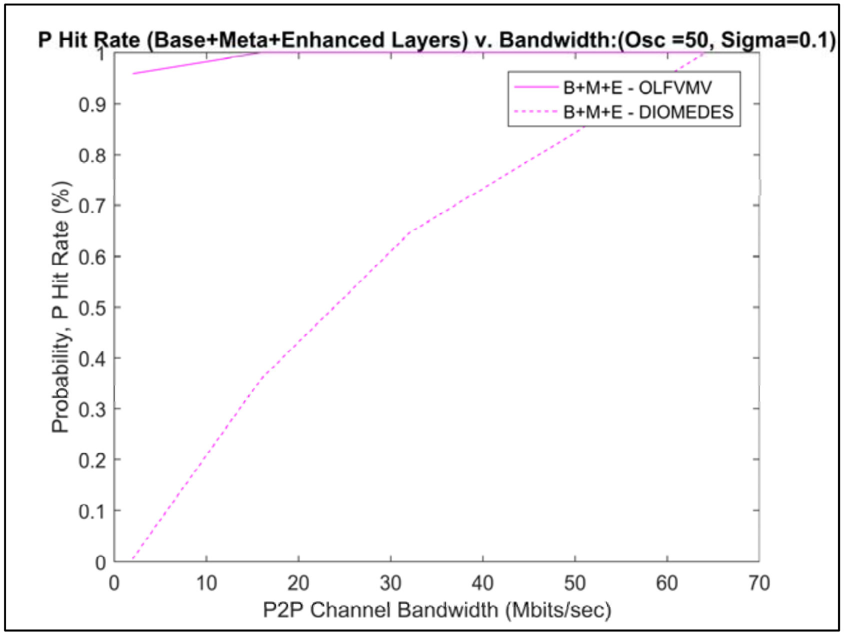


Simulation Results

For this same scenario, **DIOMEDES was not capable of transporting any of the enhanced layer content at 2 Mbps**

- **This is because the P2P channel alone was not capable of transporting the RIGHT core camera enhanced layer** (at 4 Mbps) for any viewer.

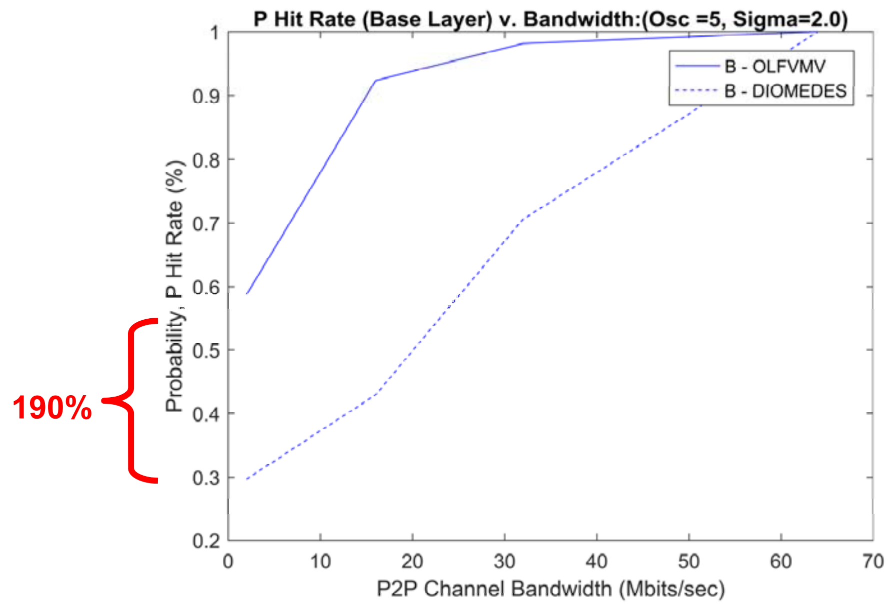




Simulation Results

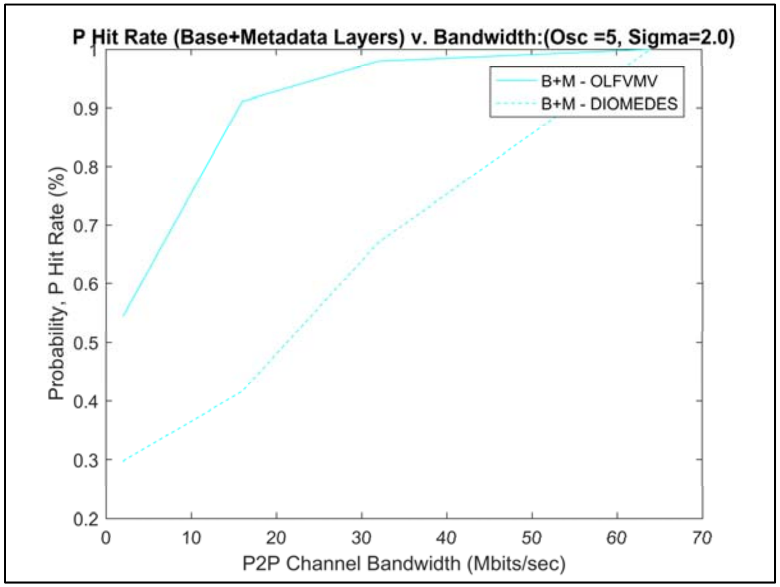
OLFVmv and DIOMEDES performance was closest to each other when the viewing pattern entailed a rapid oscillation (Osc = 5 seconds) and the randomized dispersion of viewpoints between viewer's was at the highest (sigma scale factor = 2.0)

- Nonetheless, at P2P bandwidth throughput of 2 Mbps, the performance of OLFVmv exceeded that of DIOMEDES by 190%, for the base layer

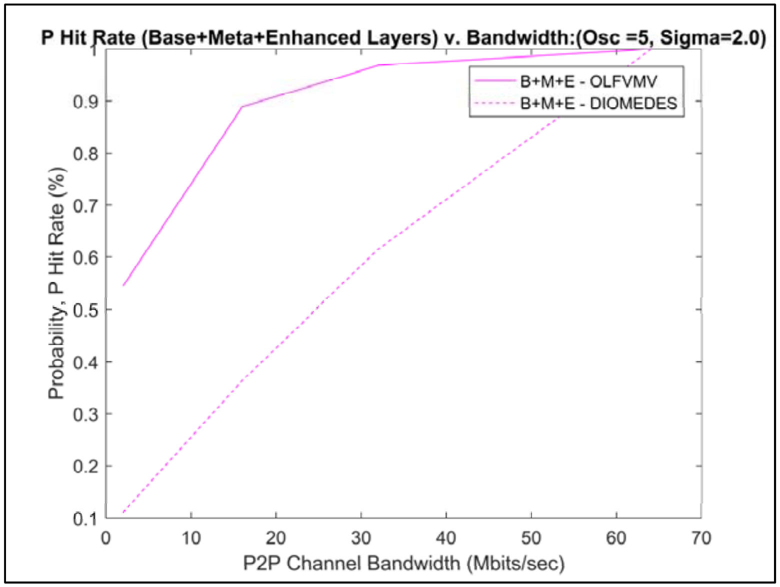


Simulation Results

179%



454%



Simulation Results - Summary

- OLFVmv's performance far exceeded the performance of DIOMEDES in all cases below 64 Mbps.
- Superior results based on OLFVmv's ability to adaptively sense and prioritize video content.
- OLFVmv provides practical bandwidth requirements.

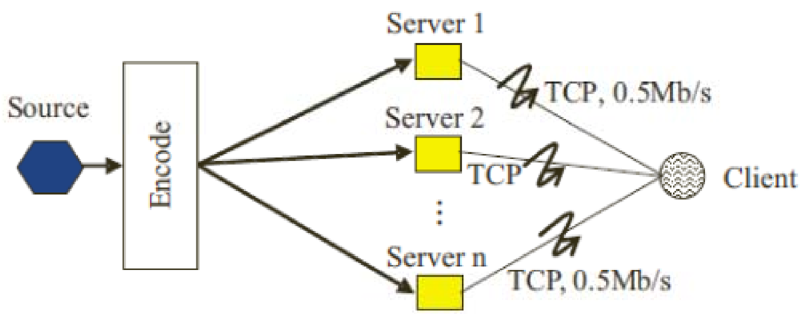
OLFVmv is important because it opens the door for the use of true live free viewpoint video using standard DVB channels augmented with a limited throughput P2P channel throughput

Agenda

- Background
  - Video Compression and SVC (Scalable Video Coding)
- Multiview Video Types
  - Multiview Coding (MVC) Types and Industry Standards
- State of the Industry – FTV (Free Viewpoint TV)
- OLFVmv (Optimized Live Free Viewpoint multiview video)
  - Motivation
  - Contribution
  - Architecture
  - Algorithms
- Simulation Results
- Extensions to OLFVmv Using Network Coding
- Further Optimization of OLFVmv – Ph.D. Thesis Work
- Conclusion

Future Work: Extensions to OLFVmv Using Hierarchical Network Coding (HNC)

- Extensions to HNC are ideally suited for OLFVmv



HNC network path diversity streaming topology showing source and intermediate nodes

- Whereas, HNC can be applied to OLFVmv's SVC layering (e.g., base, metadata, enhanced layers) and prioritization of lower priority core camera views

Layer number 1, 2 ... i

$$p_i = \sum_{j=1}^{m_1} f_j^1 b_j^1 + \sum_{j=1}^{m_2} f_j^2 b_j^2 + .. + \sum_{j=1}^{m_i} f_j^i b_j^i$$

Original packets for each layer 1, 2 … i

Non-zero random elements of finite filed F_q

HNC network packet encoding by layers

[NguyenNguyenCheung2010]

Oregon State University OSU

Future Work: Extensions to OLFVmv Using Hierarchical Network Coding (HNC) (Backup)

HNC inherently accommodates prioritization of OLFVmv layering and core camera views

COMPARE CODING SCHEMES WITH 2 LAYERS DATA

Uncoded	WLNC	Hierarchical NC	RNC
a_1	a_1	a_1	a_1
a_2	a_2	a_2	a_2
	$a_1 + a_2$	$a_1 + a_2$	$a_1 + a_2$
b_1	b_1	$a_1 + b_1$	$a_1 + b_1$
b_2	b_2	$a_1 + b_2$	$a_1 + b_2$
	$b_1 + b_2$	$a_1 + b_1 + b_2$	$a_1 + b_1 + b_2$
		$a_2 + b_1$	$a_2 + b_1$
		$a_2 + b_2$	$a_2 + b_2$
		$a_2 + b_1 + b_2$	$a_2 + b_1 + b_2$
		$a_1 + a_2 + b_1$	$a_1 + a_2 + b_1$
		$a_1 + a_2 + b_2$	$a_1 + a_2 + b_2$
		$a_1 + a_2 + b_1 + b_2$	$a_1 + a_2 + b_1 + b_2$
			b_1
			b_2
			$b_1 + b_2$

Highest priority (thus most redundant) e.g, base layer content or most important core camera views

Lower priority (thus least redundant) e.g., enhancement layer content or least important core camera views

[NguyenNguyenCheung2010]

Future Work: Extensions to OLFVmv Using Hierarchical Network Coding (HNC) (Backup)

Example:

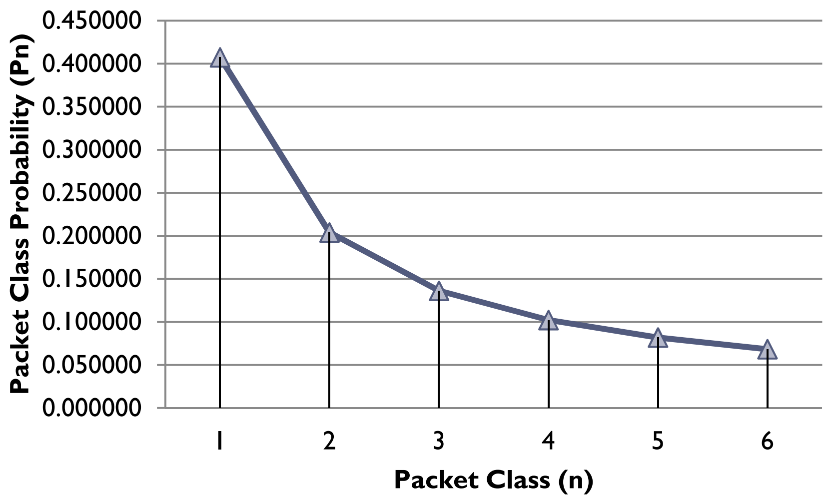
$$pl = \sum_{n=1}^{MAX\_CORE\_CAMERAS} \left(\sum_{j=1}^{m\_META} f_{n,j}^{META}\, b_{n,j}^{META} + \sum_{j=1}^{m\_BASE} f_{n,j}^{BASE}\, b_{n,j}^{BASE} + \sum_{j=1}^{m\_ENHANCED} f_{n,j}^{ENHANCED}\, b_{n,j}^{ENHANCED} \right)$$

Packet Class (n)	Packet Class Probability (Pn)	Hierarchical NC for FTV Video Content
1	P_1	$V3_{Base}$ $V3_{Meta}$ $V3_{Enhanced}$ $V3_{Base} + V3_{Meta}$ $V3_{Base} + V3_{Enhanced}$ $V3_{Meta} + V3_{Enhanced}$ $V3_{Base} + V3_{Meta} + V3_{Enhanced}$
2	P_2	$V3_{Base} + V6_{Base}$ $V3_{Base} + V6_{Meta}$ … $V3_{Enhanced} + V6_{Base}$ … $V3_{Base} + V3_{Meta} + V6_{Base}$ … $V3_{Base} + V3_{Meta} + V3_{Enhanced} + V6_{Base} + V6_{Meta} + V6_{Enhanced}$
…	…	…
12	P_6	… $V3_{Base} + V3_{Meta} + V3_{Enhanced} + V6_{Base} + V6_{Meta} + V6_{Enhanced} + V2_{Base} + V2_{Meta} + V2_{Enhanced} +$ $V7_{Base} + V7_{Meta} + V7_{Enhanced} + V1_{Base} + V1_{Meta} + V1_{Enhanced} + V8_{Base} + V8_{Meta} + V8_{Enhanced}$
SUM	**= 1**	

Future Work: Extensions to OLFVmv Using Hierarchical Network Coding (HNC)(Backup)

$$Pn = \frac{\dfrac{C!}{(n/(C!-n))}}{\sum_{n=1}^{C} \dfrac{C!}{(n/(C!-n))}}$$

Expression of P2P packet priorities within chunks
(C = total number of packet classes)



Example - Hierarchical Network Coding (HNC) packet class priorities as applied to FTV core camera video content distribution over the P2P channel

Packet Class (n)	Packet Class Probability (Pn)
1	0.407339
2	0.203953
3	0.136158
4	0.102261
5	0.081923
6	0.068365
SUM	**1.000000**

Agenda

- Background
  - Video Compression and SVC (Scalable Video Coding)
- Multiview Video Types
  - Multiview Coding (MVC) Types and Industry Standards
- State of the Industry – FTV (Free Viewpoint TV)
- OLFVmv (Optimized Live Free Viewpoint multiview video)
  - Motivation
  - Contribution
  - Architecture
  - Algorithms
- Simulation Results
- Extensions to OLFVmv Using Network Coding
- Further Optimization of OLFVmv – Ph.D. Thesis Work
- Conclusion

Further Optimization of OLFVmv – Ph.D. Thesis Work

An opportunity exists to ***improve viewing trend prediction algorithms to enhance video content selection and prioritization***

Objective: Improved Prediction Modeling/Algorithms Through Reinforcement Learning

Can our OLFVmv system learn how to best optimize video content prioritization?

Oregon State UNIVERSITY OSU

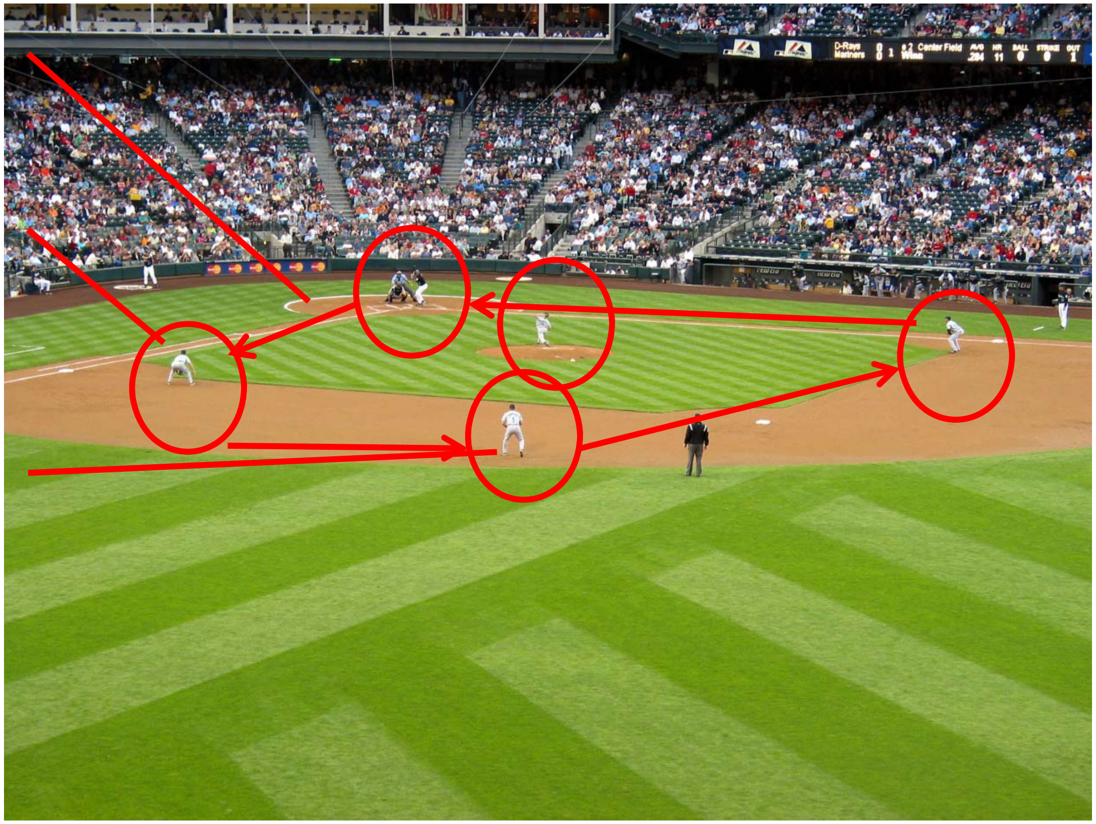
Further Optimization of OLFVmv – Ph.D. Thesis Work

By applying Markov Decision Processes / Reinforcement Learning we can teach the OLFVmv system to optimize video content for ***future states***

And various trends

Based on some observed state S

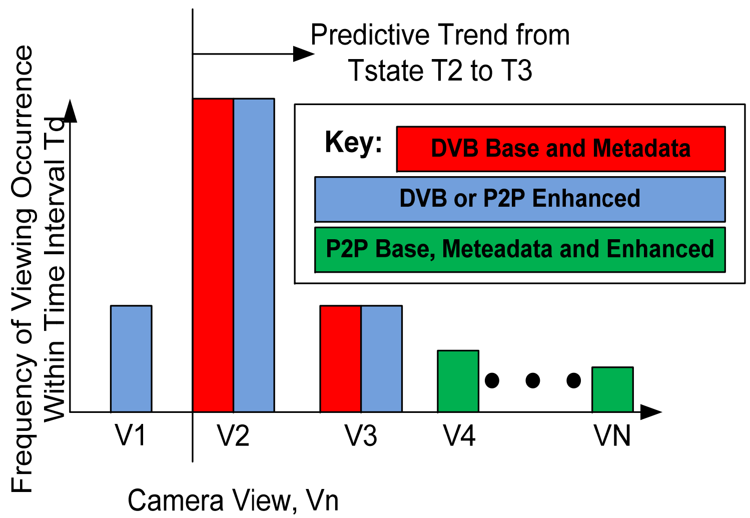
And some other less popular observed states



Further Optimization of OLFVmv – Ph.D. Thesis Work

Overall Objectives Deliverables:

1) Develop machine learning policies
2) Emulation on a virtual machine
3) Theoretical reconstruction of content packets



OLFVmv viewing trend modeling

Oregon State University OSU

Agenda

- Background
  - Video Compression and SVC (Scalable Video Coding)
- Multiview Video Types
  - Multiview Coding (MVC) Types and Industry Standards
- State of the Industry – FTV (Free Viewpoint TV)
- OLFVmv (Optimized Live Free Viewpoint multiview video)
  - Motivation
  - Contribution
  - Architecture
  - Algorithms
- Simulation Results
- Extensions to OLFVmv Using Network Coding
- Further Optimization of OLFVmv – Ph.D. Thesis Work
- **Conclusion**

Conclusion

OLFVmv is proven to provide:

- ✓ A well defined, practical transport of FTV over ***existing broadcast*** mediums[1]:
  - ✓ **Using normal DVB** bandwidths
  - ✓ **Using only 2 DVB channels**
- ✓ A practical transport of FTV over P2P networks:
  - ✓ **Enables low bandwidth, mobile P2P** networks using as little as 2 Mbps
- ✓ **Superior performance** over other proposed FTV transport means

[ETSI EN300 429 V1.2.1:1998][ETSI EN200 421 V1.1.2:1997][ETSI EN302 307:2009]
[ETSI EN302 304 V1.1.1:2004][ETSI TS102 034 V1.4.1:2009][ETSI EN 302 755 V1.2.1:2010]

References

[Cheung2011] Cheung, et al., "Interactive Streaming of Stored Multiview Video Using Redundant Frame Structures", *IEEE Transactions on Image Processing*, Vol. 20, No. 3, 744-761, March 2011.

[DIOMEDES D2.3 2012] DIOMEDES standard D2.3, "Final reference system architecture report", Final Revision, dated Jan 31, 2012.

[DIOMEDES D3.6 2011] DIOMEDES standard D3.6, "Public report on 3D Audio / Video rendering and content adaption", Final revision, dated 10/31/2011.

[DIOMEDES D4.4 2011] DIOMEDES standard D4.4, "Report on the developed audio and video codecs", Final revision, Oct 31, 2011.

[DIOMEDES D4.5 2011] DIOMEDES standard D4.5, "Report on results of integrated MD-SMVD and P2P system", Final revision, dated 12/31/2011.

[DIOMEDES_Flyer2013] "Distribution of MultiView Entertainment Using Content Aware Delivery Systems, 2013.

[Dufaux2013] Frederic Dufaux et al., "Emerging Technologies for 3D Video: Creation, Coding, Transmission and Rendering", *Wiley Publishing*, 2013.

[ETSI EN300 421 V1.1.2:1997] ETSI EN 300 421 V1.1.2, "Digital Video Broadcasting (DVB); Framing structure, channel coding and modulation for 11/12 GHz satellite services", August, 1997.

[ETSI EN300 429 V1.2.1:1998] ETSI EN 300 429 V1.2.1, "Digital Video Broadcasting (DVB); Framing structure, channel coding and modulation for cable systems", April, 1998.

[ETSI EN302 304 V1.1.1:2004] ETSI EN 300 304 V1.1.1, "Digital Video Broadcasting (DVB); Transmission System for Handheld Terminals (DVB-H)", November, 2004.

Oregon State UNIVERSITY OSU

References

[ETSI EN302 307 V1.2.1:2009] ETSI EN 302 307 V1.2.1, "Digital Video Broadcasting (DVB); Second generation framing structure, channel coding and modulation systems for Broadcasting, Interactive Services, News Gathering and other broadband satellite applications (DVB-S2)", August, 2009.

[ETSI EN302 755 V1.2.1:2010] ETSI EN 302 755 V1.2.1, "Digital Video Broadcasting (DVB); Frame structure channel coding and modulation for a second generation digital terrestrial television broadcasting system (DVB-T2)", October, 2010.

[ETSI TS102 034 V1.4.1:2009] ETSI TS 102 034 V1.4.1, "Digital Video Broadcasting (DVB); Transport of MPEG-2 TS Based DVB Services over IP Based Networks", August, 2009.

[InterFrameCompression] Interframe compression estimates, at link: https://en.wikipedia.org/wiki/Inter_frame, last visited Nov. 13, 2016

[ISO/IEC 14496-10:2008] ISO/IEC 14496-10:2008, "Information technology -- Coding of audio-visual objects -- Part 10: Advanced Video Coding", 2008.

[ISO/IEC 14496-10:2009] ISO/IEC 14496-10:2009, "Information technology -- Coding of audio-visual objects -- Part 10: Advanced Video Coding" extension, including the Multiview High profile, March 16, 2009.

[ISO/IEC 14496-10:2010] ISO/IEC 14496-10:2010, "Information technology -- Coding of audio-visual objects -- Part 10: Advanced Video Coding", amendment containing definition the Multiview Stereo High profile for two-view video coding with support of interlaced coding tools and specifying an additional SEI message (the frame packing arrangement SEI message), March 9, 2010.

[ISO/IEC 14496-10:2014] ISO/IEC 14496-10:2014, "Information technology -- Coding of audio-visual objects -- Part 10: Advanced Video Coding", amendment that specifies the coding of depth map data for 3D stereoscopic video, including a Multiview Depth High profile.), August 27, 2014

References

[KimLee2011] Young-il Kim, Yong Su Lee, et al., "The Performance Analysis of SVC image Service for Mobile IPTV System", 13th International Conference on Advanced Communication Technology (ICACT), 2011, pages 1142-1145, February 13-16, 2011.

[Kramer2016] Richard A. Kramer, "An Introduction to the Problem: Interactive Free Viewpoint Live Multiview Video Streaming Using Network Coding", Oregon State University, 2016.

[Morvan_deWithFarin2006] Yannick Morvan, Peter H. N. de With and Dirk Farin, "Platelet-based coding of depth maps for the transmission of multiview images", Eindhoven University, 2006

[MüllerSchwarz2013] Karsten Müller, Heiko Schwarz, et al., "3D High-Efficiency Video Coding for Multi-View Video and Depth Data", IEEE Transactions on Image Processing, Vol. 22, No. 9, pages 3366-3378, September 2013.

[OhmSullivan2005] Jens-Rainer Ohm, Gary Sullivan, "MPEG-4 Advanced Video Coding", MPEG doc#: N7314, July 2005

NguyenNguyenCheung2010] Kien Nguyen, Thinh Nguyen, and Sen-Ching Cheung, "Video Streaming with Network Coding", Journal of Signal Processing Systems Archive Volume 59 Issue 3 at pages 319-333, June 2010.

[Ohm2009] Jens-Rainer Ohm, "MPEG Developments in Multi-view Video Coding and 3D Video", *Multiview & 3D Video Coding – EBU Workshop*, April 2009.

[Polycom2010] Polycom Whitepaper, "More Scale at Lower Cost with Scalable Video Coding", November 2010

[Rimac-DjljeNemčićVranješ2008] Snježana Rimac-Drlje1, Ognjen Nemčić, Mario Vranješ, "Scalable Video Coding Extension of the H.264/AVC Standard", 50th International Symposium ELMAR-2008, pgs. 9-12, September 10-12, 2008.

[SchierlNarasimhan2011] Thomas Schierl, Sam Narasimhan, "Transport and Storage Systems for 3-D Video Using MPEG-2 Systems, RTP, and ISO File Format", Proceedings of the IEEE, Vol. 99, No 4, pages 671-683, April, 2011.

References and Future Reading

[Smolic2008] Aljoscha Smolic, "Advanced Video Coding" subsection "Multiview Video Coding", MPEG-4, The Motion Picture Experts Group, MPEG doc #N9580, January 2008, June 16, 2016 article was accessed at: http://mpeg.chiariglione.org/standards/mpeg-4/advanced-video-coding.

[Schwarz2013] Dr.-Irg. Heiko Schwarz, "Source Coding and Compression", December 7, 2013, December 12, 2016 article was access at: iphome.hhi.de/schwarz/assets/pdfs/07_TransformCoding.pdf.

[UittoVehkapera2013] Mikko Uitto, Janne Vehkapera, "Enhanced Quality Adaptation Strategies for Scalable Video", Signal Processing and Information Technology(ISSPIT), 2013 IEEE International Symposium, pages 74-79, 2013

[VetroWiegandSullivan2011] Anthony Vetro, Thomas Wiegand and Gary Sullivan, "Overview of the Stereo and Multiview Video Coding Extensions of the H.264/MPEG-4 AVC Standard", *Proceedings of the IEEE*, Vol. 99, No. 4, at pages 626-642, 2011.

[Mitrovic] Djordje Mitrovic, "Video Compression", University of Edinburgh, undated, December 12, 2016 article was accessed at: homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/AV0506/s0561282.pdf.

[VideoBandwidthEstimates] Video bandwidth estimates, at link: http://mathscinotes.com/2012/05/high-definition-television-bandwidth-and-compression-math/, last visited Nov. 13, 2016

[WiegandSullivan2003] Thomas Wiegand, Gary J. Sullivan, "Overview of the H.264/AVC Video Coding Standard" IEEE Transactions on Circuits and Systems for Video Technology, Vol. 13, No. 7, July 2003

Questions?

Thank you!