Multiple transforms for video coding

Adrià Arrufat

Orange Labs and IETR/INSA — Rennes, France

11th December 2015

Director: Olivier Déforges

Supervisor: Pierrick Philippe

École Doctorale Matisse

Outline

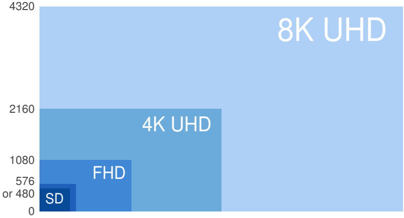
Era of the Internet and videos

Current situation

- In 2014: around 70% of Internet traffic was due to video streaming
- Forecast for 2019: more than 80%[1]

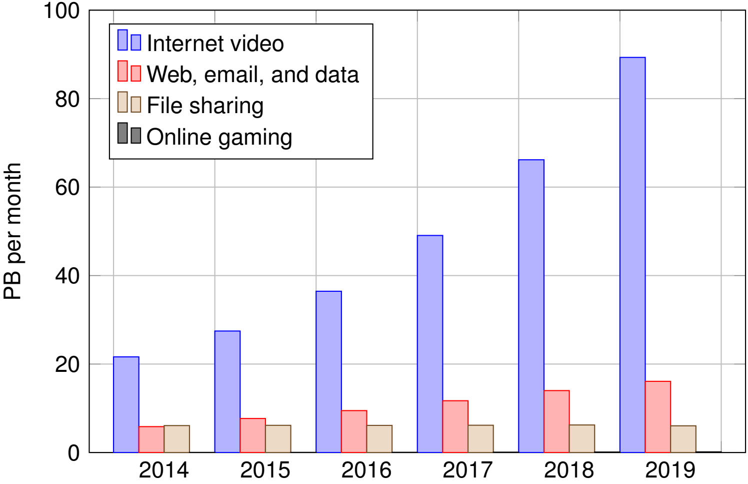
Continuous need for video compression

- **New formats** emerge
- New applications require **improved video quality**
- Need to decrease the bit-rate to stream/store videos



[1] Cisco Visual Networking Index: Forecast and Methodology, 2014–2019 White Paper

Era of the Internet and videos



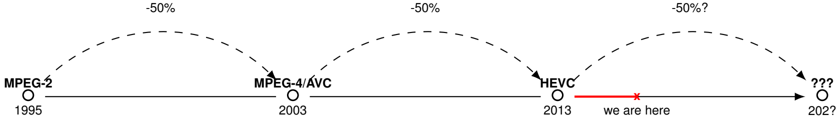
Standardisation

Video coding standards

- The work is inscribed inside a standardisation context
- Latest standard, HEVC, was released in 2013

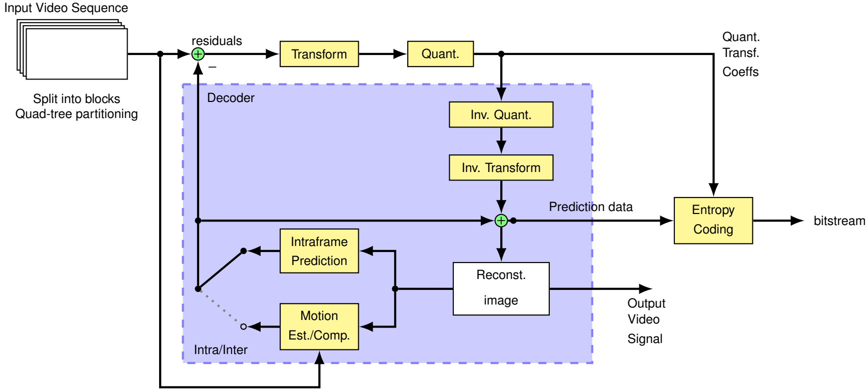
Working context

- Beginning of the standardisation phase
- Exploratory phase with relaxed complexity constraints
- Goal: achieve a suitable solution for a video coding standard for around 2020.



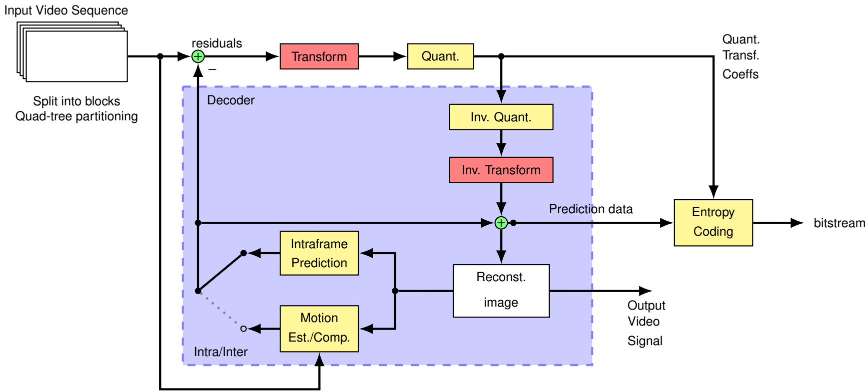
The hybrid video coding scheme

Used in most video coding standards



The hybrid video coding scheme

Used in most video coding standards



Outline

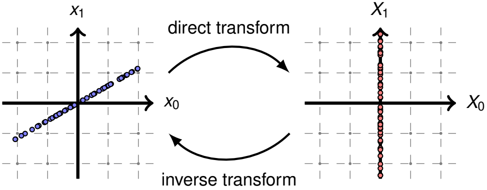
Introduction to transforms

Definition

- A transform is a mathematical function of a signal from a representation domain to another, e.g. a rotation (2D)
- A transform is a change of basis

Desirable properties

- Low complexity
  - real time
  - battery drain
- Compact representation
- Orthogonality
  - perfect reconstruction
  - easily invertible



Separability and non-separability

Pros & Cons

Non-separable

- Able to exploit any linear correlation within a block
- They require $\approx N^4$ operations

Separable

- Able to decorrelate pixels sharing rows or columns
- They require $\approx 2N^3$ operations

0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15

Example with $N = 4$

Separability and non-separability

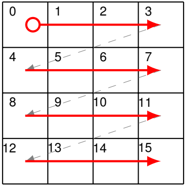
Pros & Cons

Non-separable

- Able to exploit any linear correlation within a block
- They require $\approx N^4$ operations

Separable

- Able to decorrelate pixels sharing rows or columns
- They require $\approx 2N^3$ operations



Example with $N = 4$

Separability and non-separability

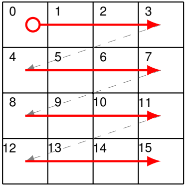
Pros & Cons

Non-separable

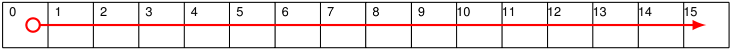
- Able to exploit any linear correlation within a block
- They require $\approx N^4$ operations

Separable

- Able to decorrelate pixels sharing rows or columns
- They require $\approx 2N^3$ operations



Example with $N = 4$



What kind of data is processed by transforms?

Particular case of HEVC

Residual blocks

- Difference between:
  - original block
  - predicted block
- Examples:

- The same transform is used for all of them

They are generated as different combinations of

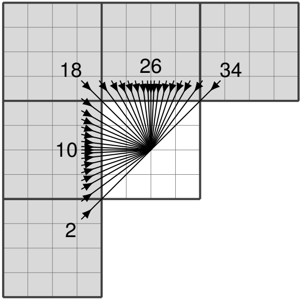
- transform units (TUs): 32, 16, 8, 4
- predictions
  - spatial or intra: 0,1,. . . ,34
  - temporal or inter

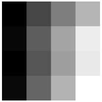
The choice is made by the encoder

- it selects the best combination in terms of:
  - distortion
  - rate

The intra prediction scheme

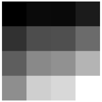
Intra Prediction Modes (IPMs)





Average residual for IPM 10

Average residual for IPM 18



Average residual for IPM 26

How are transforms conceived?

The trade-off

$J(\lambda) = \text{Distortion} + \lambda \text{ Rate}$

State of the art

- KLT → Optimal transform assuming
  - large amount of bits → High-resolution quantisation hypothesis
- Hypothesis for optimality is not valid in video coding
- RDOT → Alternative design approach[1]
  - focused on signal sparsity in the transform domain

References

[1] O.G. Sezer — Sparse orthonormal transforms for image compression, 2008

The rate-distortion optimised transform

RDOT design features

- Minimise the distortion and the number of significant coefficients
- Use a simple model for the bit-rate
  - ℓ_0 norm: number of non-zero coefficients
- Sparsity suits video coding syntax elements

The RDOT equation

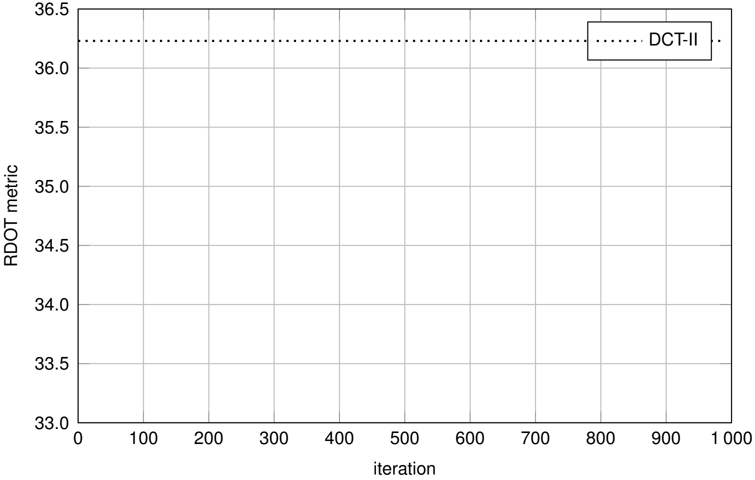
$$J(\lambda) = \sum_{\forall i} \underbrace{\left\| \mathbf{x}_i - \mathbf{A}^T \cdot \mathbf{c}_i \right\|^2}_{\text{Distortion}} + \lambda \underbrace{\left\| \mathbf{c}_i \right\|_0}_{\text{rate}} \quad \text{with} \begin{cases} \mathbf{x}_i & \text{residuals} \\ \mathbf{A} & \text{transform} \\ \mathbf{c}_i & \text{transf. \& quant. coeff.} \\ \lambda & \text{Lagrange multiplier (PDF independent)} \end{cases}$$

- iterative algorithm

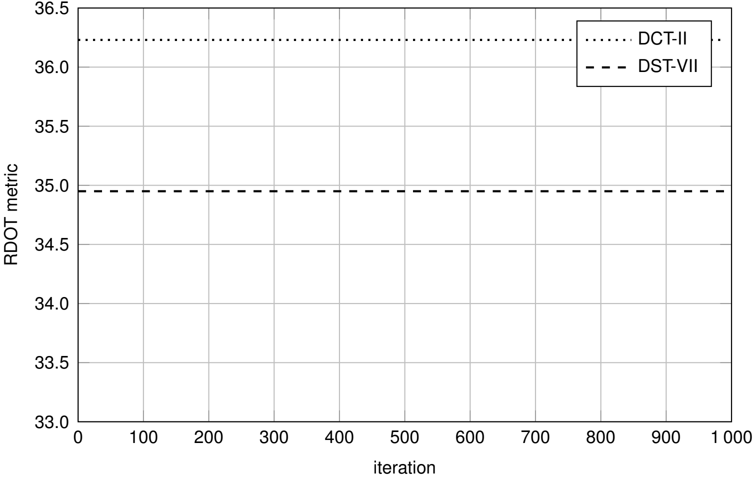
Objective

- For a given set of residuals $\mathbf{x}_i$'s
- Find the transform $\mathbf{A}$ that minimises $J(\lambda)$

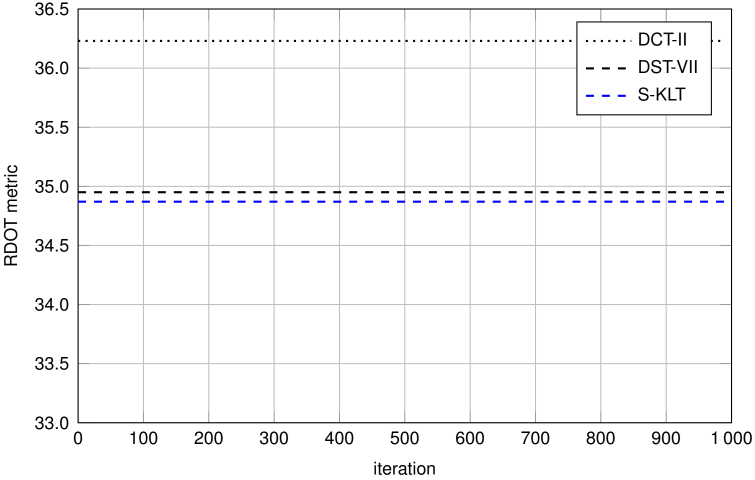
RDOT metric for different transforms on 4×4 residuals



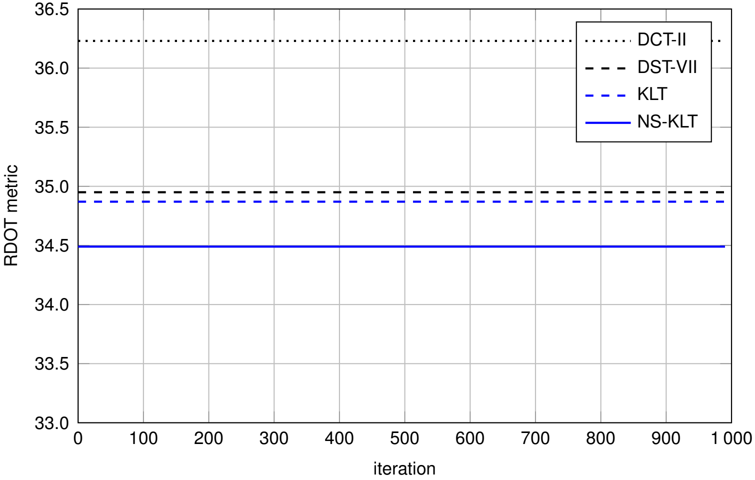
RDOT metric for different transforms on 4×4 residuals



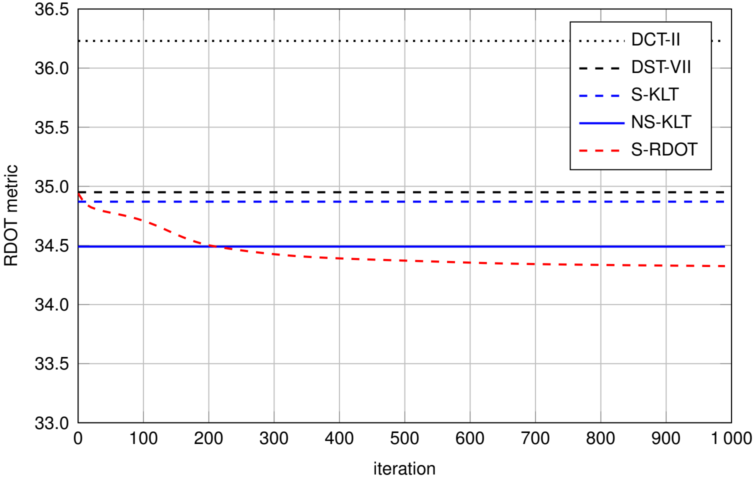
RDOT metric for different transforms on 4×4 residuals



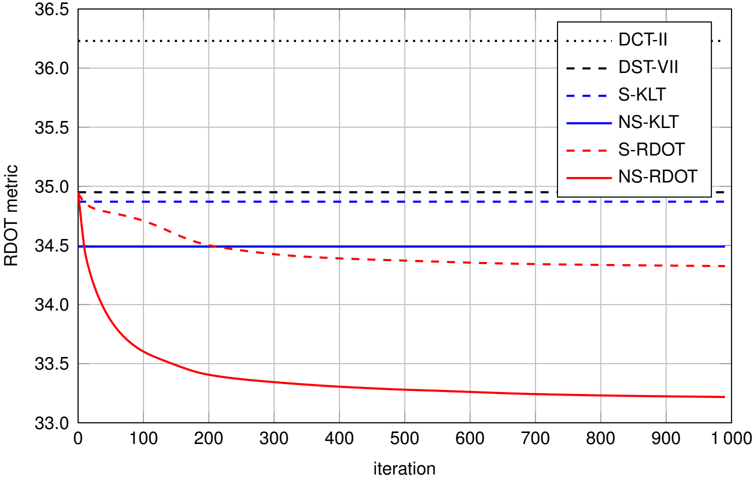
RDOT metric for different transforms on 4×4 residuals



RDOT metric for different transforms on 4×4 residuals



RDOT metric for different transforms on 4×4 residuals



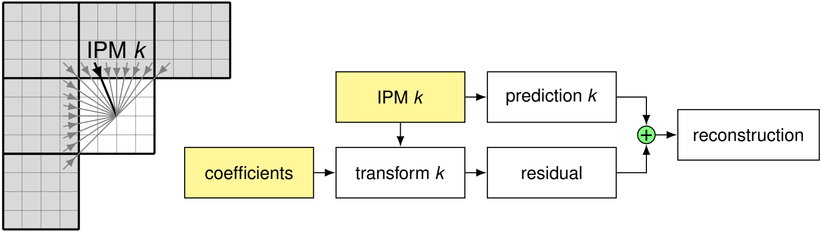
Experiment: Using the transforms in HEVC

Mode-dependent directional transforms (MDDT) for 4 × 4 and 8 × 8 TUs

Learning phase

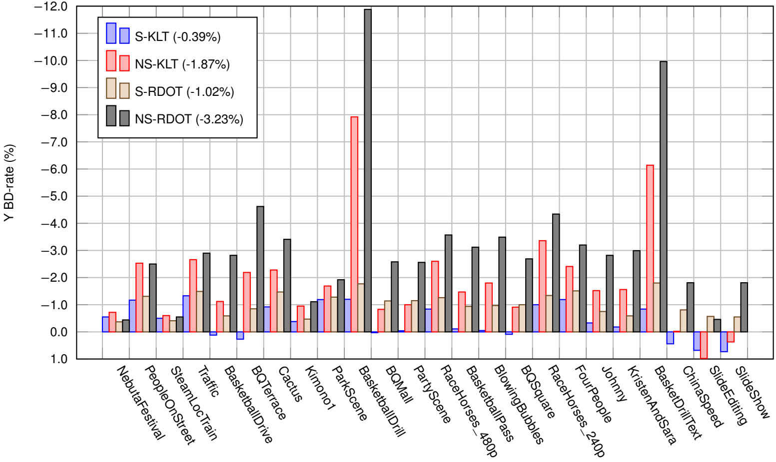
- Learn an adapted transform to each of the 35 IPMs (MDDT)
- Replace the default HEVC transforms (no additional signalling) with:
  - KLT
  - RDOT

Decoding scheme



MDDT performances on the HEVC test set

Y BD-rate (%) on AI for 4×4 and 8×8 blocks



Conclusions

Performance over HEVC

	KLT		RDOT	
	sep	non-sep	sep	non-sep
Y BD-rate	-0.39%	-1.87%	-1.02%	-3.23%
Encoding	108%	112%	108%	112%
Decoding	105%	120%	105%	120%
ROM	8.20 kB	148.75 kB	8.20 kB	148.75 kB

Transform design

- RDOT design provides significant better results than the KLT design
- Separability plays an important role on the performance
  - complexity (encoding/decoding)
  - storage requirements
  - bit-rate savings

Next steps

- Find out the limits of this technique

Outline

Mode-dependent transform competition (MDTC)

Motivations

- Residual variability even inside the same IPM
- Significantly better results using RDOTs in MDDT than KLTs

Multiple transform design for HEVC

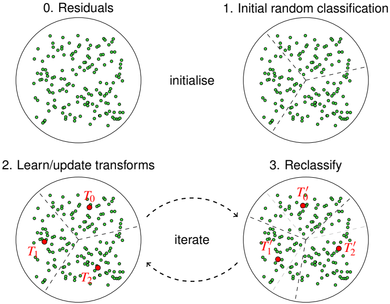
- Conservative approach → $1 + 2^N$ transforms
  - default transform (DCT/DST) + additional RDOTs
  - signalling → flag plus fixed length codeword
- Learning algorithm → based on the RDOT metric

Multiple transform design

Classic clustering method: classify/update

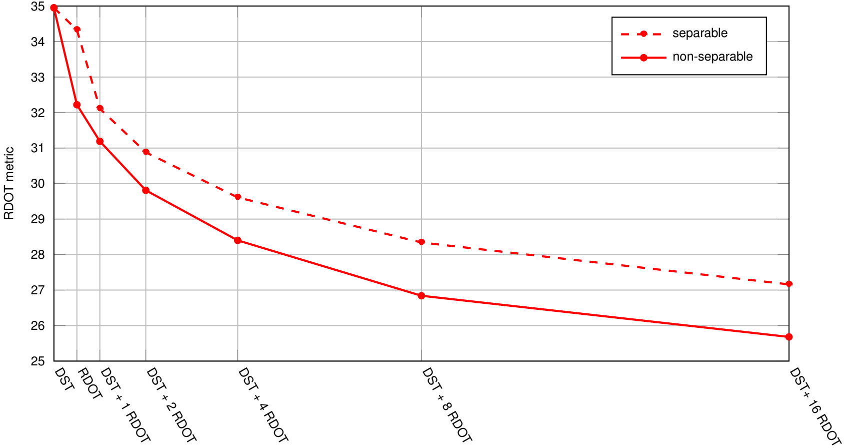
Using the RDOT metric

- $J(\lambda) = \sum_{\forall i} \left\| \mathbf{x}_i - \mathbf{A}^T \cdot \mathbf{c}_i \right\|^2 + \lambda \left\| \mathbf{c}_i \right\|_0$
- It is used to evaluate a transform
  - compute the optimal transform for a set of residuals
  - assign each residual to the transform that minimises $J(\lambda)$
- It allows creating an iterative clustering algorithm



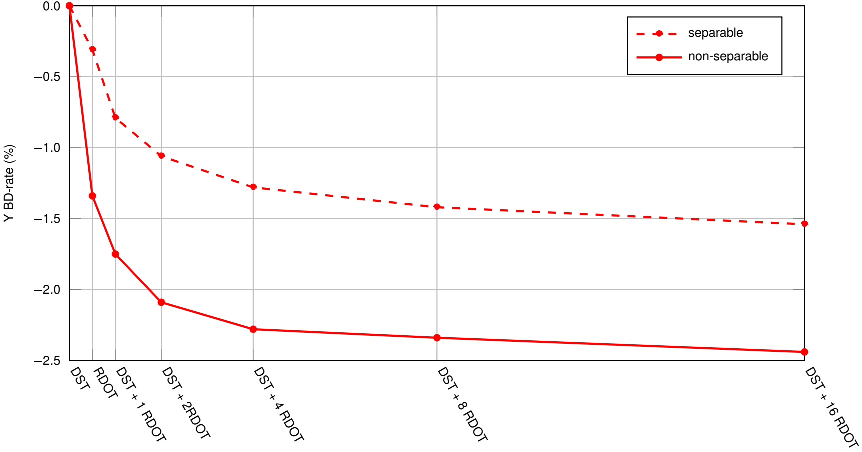
Learning results

Averaged RDOT metric for 4 × 4 blocks across all IPMs



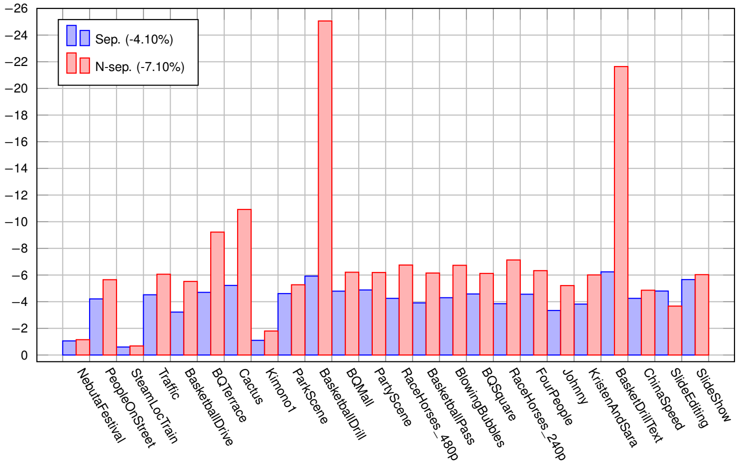
Results in video coding on top of HEVC

Y BD-rate (%) for 4×4 blocks



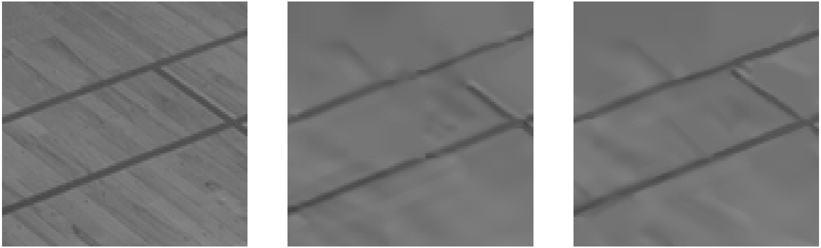
MDTC performances in detail

Y BD-rate (%) for High Performance System: $4 \times 4 : 1 + 16$ & $8 \times 8 : 1 + 32$



MDTC performances in detail

Graphical improvement at QP 37 on BasketballDrill (3 Mbps)



(a) Original (b) HEVC (c) MDTC

Figure: Comparison between HEVC and MDTC

Conclusions

Performances

	4×4: 1+16	8×8: 1+32
	sep.	non-sep.
Y BD-rate	-4.10%	-7.10%
Enc. Time	800%	2000%
Dec. Time	105%	120%
ROM	236.25 kB	4.51 MB

Next steps

- Important bit-rate savings thanks to transform competition
- Non-separable transforms are very complex
  - encoding/decoding complexity
  - storage requirements
- Simplify MDTC systems to make them usable

Outline

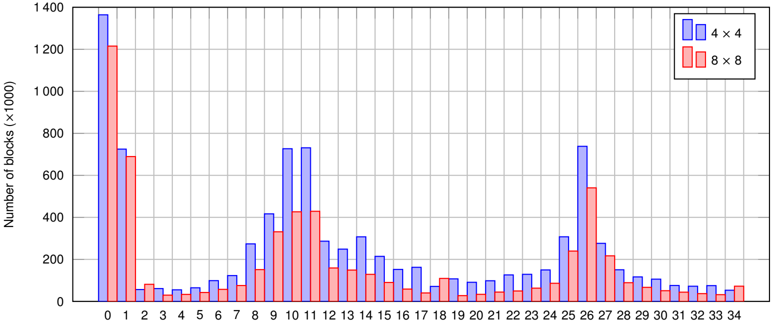
Outline

1. General introduction
2. Transform coding
3. Mode-dependent transform competition
4. Simplifications of the MDTC systems
   - ROM limitations on MDTC systems
   - MDTC systems based on Discrete Trigonometric Transforms
5. Conclusions

Different number of transforms per IPM

Observations

- Proposed MDTC uses the same number of transforms in all IPMs
- IPM usage is not uniform: there are some favoured modes (MPM)

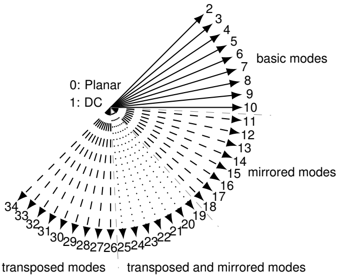


Symmetries in intra prediction residuals

Observations

- Residuals coming from specific IPMs present geometrical relations
- With proper manipulation, transforms can be reused

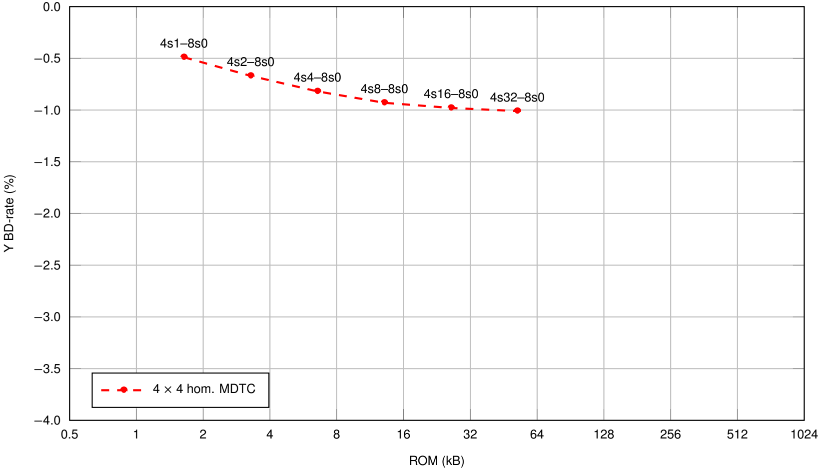
Graphical interpretation



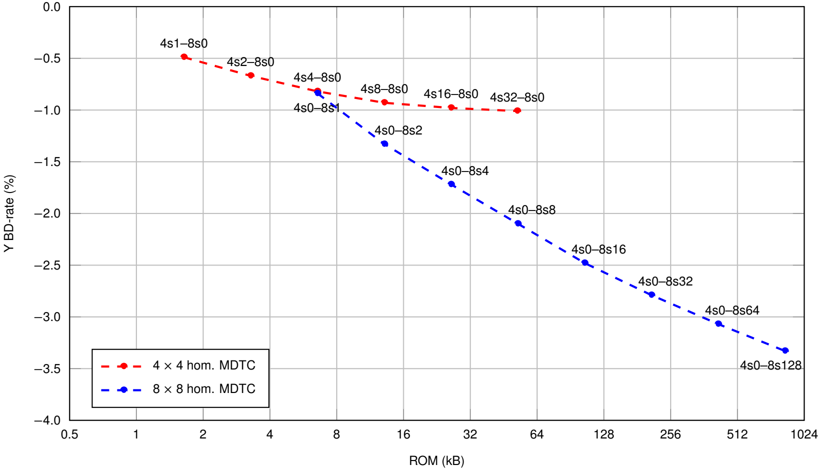
Operations applied to blocks

$$\mathbf{X} = \begin{cases} \mathbf{A}\,\mathbf{x} & 0 \leq \text{IPM} \leq 10 \\ \mathbf{A} \circlearrowleft \mathbf{x} & 11 \leq \text{IPM} \leq 18 \\ \mathbf{A} \circlearrowleft \mathbf{x}^T & 19 \leq \text{IPM} \leq 25 \\ \mathbf{A}\,\mathbf{x}^T & 26 \leq \text{IPM} \leq 34 \end{cases}$$

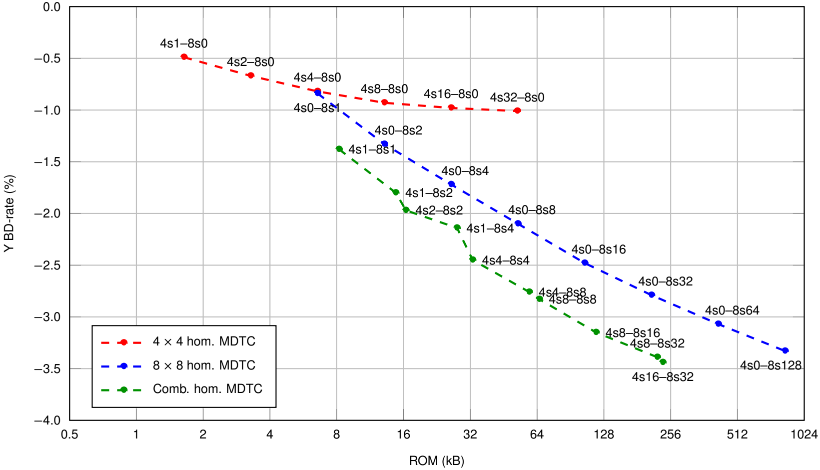
Results on the ROM — BD-rate plane



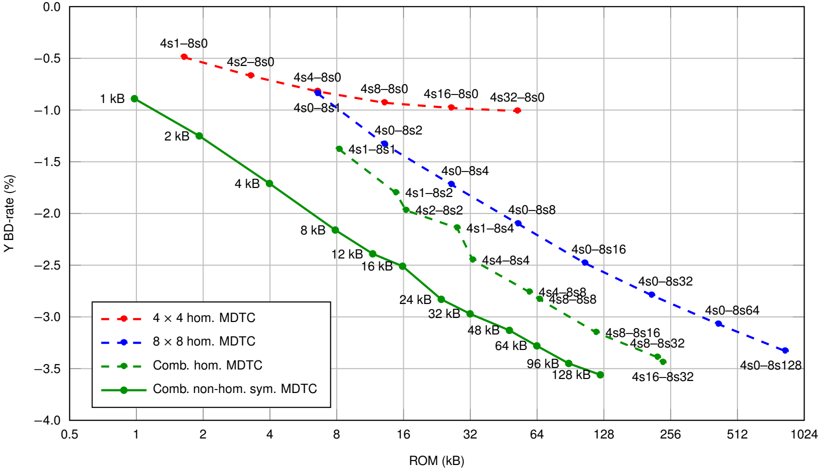
Results on the ROM — BD-rate plane



Results on the ROM — BD-rate plane



Results on the ROM — BD-rate plane



Summary and conclusions

Systems on demand (performances on the HEVC test set)

System	16 kB	32 kB	64 kB	128 kB
Y BD-rate	-2.87%	-3.21%	-3.55%	-3.79%
Encoding	372%	481%	761%	1297%
Decoding	105%	105%	105%	105%

Remarkable points

- ROM can be reduced up to a third of its original value
- Bit-rate savings are maintained
- Coding complexity remains more or less untouched

Outline

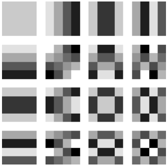
Discrete Trigonometric Transforms (DTTs)

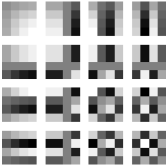
DCTs and DSTs

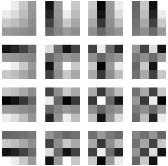
DTT strengths

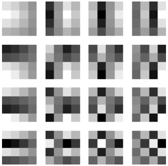
- Fast algorithms
  - number of operations in the order of $N \log_2 N$ instead of N^2
- Transform coefficients can be computed analytically
  - notable reduction in storage requirements
- They can be more easily pushed into a standard

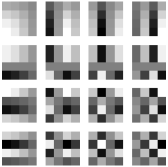
Examples of 4×4 DTTs

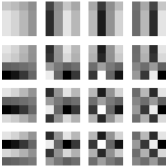

(a) **DCT-II — DCT-II**

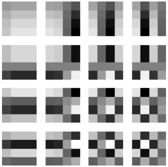

(b) **DST-VII — DST-VII**

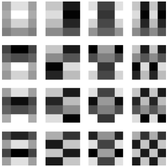

(c) DCT-IV — DCT-IV


(d) DCT-III — DCT-IV


(e) DST-VII — DCT-IV


(f) DST-V — DCT-IV


(g) DST-VII — DCT-V


(h) DST-VII — DST-II

Designing a DTT-based MDTC system

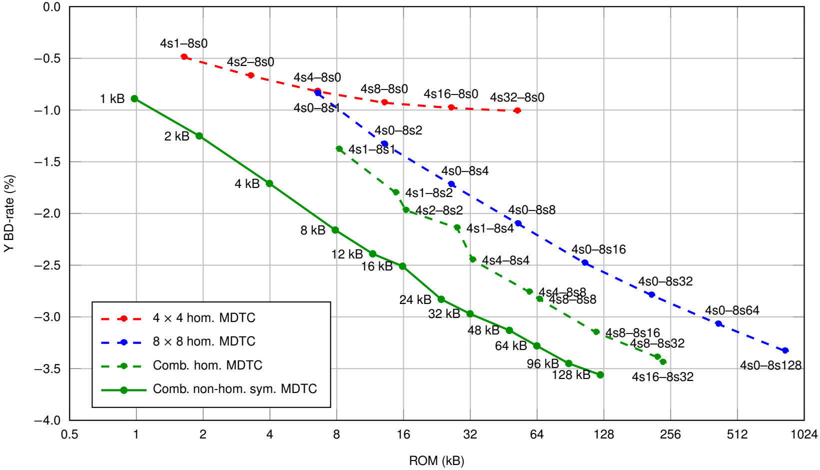
Using the RDOT metric to classify

- Transforms are already known (8 DCTs, 8 DSTs and their inverses)
- There are 256 possible transforms
- Complexity of the classifying algorithm:
  - selection the best combination of N transforms amongst 256
  - example for $N = 4$:

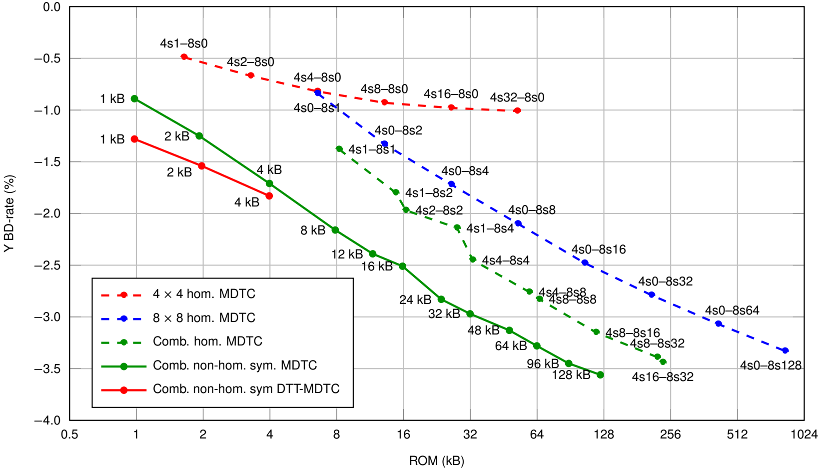
$$\binom{256}{4} = 174\,792\,640 \approx 1.75 \times 10^{8} \tag{1}$$

- The algorithm suboptimality increases with N
- Symmetries and non-homogeneous repartition of transforms are used with DTTs

Performances of DTT-based MDTC systems



Performances of DTT-based MDTC systems



Summary

DTT-based MDTC system compared to RDOT-based ones

	DTT			RDOT		
System	1 kB	2 kB	4 kB	1 kB	2 kB	4 kB
Y BD-rate	-1.28%	-1.54%	-1.86%	-0.89%	-1.25%	-1.71%
Encoding	164%	177%	230%	126%	150%	175%
Decoding	100%	100%	100%	105%	105%	105%

Remarkable points

- Almost 2% bit-rate savings with low complexity
- Performances could be higher by improving the learning algorithm
- DTTs make transforms for bigger block sizes more reasonable
  - room for improvement when using 16×16 or even 32×32 transforms

Outline

Conclusions

Interest of multiple transforms for video coding

- This technique alone is able to achieve significant bit-rate savings over HEVC, depending on complexity:
  - non-separable transforms: up to 7%
  - separable transforms: up to 4%
  - DTTs: 2% (in progress)
  - reminder: HEVC improves intra coding by 22% over AVC
- Systematic on demand design with the RDOT metric
  - the ℓ_0 has proved to be a robust rate approximation
  - independence from residuals PDF has been proved

Perspectives and future work

Immediate action points

- extend the system
  - bigger TU sizes
  - inter coding (gain in RA is about 2/3 of AI)
  - chroma
- coding complexity
  - incomplete transforms
  - exhaustive search might not be necessary
- transform signalling and quantisation

Multiple transforms in future video formats

- Multiple transforms are part of the preliminary draft for a future video coder by ITU and MPEG

Publications

Conference papers

- 2 VCIP 2014 presented papers
- 2 ICIP 2015 presented papers
- 1 ICASSP 2016 submitted paper

Patents

- 5 filled patent applications

Thank you for your attention

Questions?

Adrià Arrufat adria.arrufat@protonmail.ch

INSTITUT NATIONAL DES SCIENCES APPLIQUÉES RENNES