Multiscale convolutional neural networks (MSCNN) for in-loop video restoration

Kiran Misra, Andrew Segall, Byeongdoo Choi

Outline of presentation

Motivation

Introduction of multiscale CNN approach

Training and Evaluation

Experimental Results

Learnings from Ablation Study

Summary

MOTIVATION

Background

In-loop filtering shows interesting coding efficiency gains:

- AV1 Loop Restoration: 2.31% (Random Access)
- VVC Adaptive Loop Filter: 4.35% (Random Access)

Residual Convolutional Neural Networks (CNNs) can provide additional gains beyond the approaches listed above (> 5%)

Unfortunately, these CNN gains are accompanied by significant increase in the number of Multiply-Accumulate (MAC) operations. Here, we focus on reducing MAC per pixel count.

MULTISCALE CNN APPROACH

prime video

Introduction

In this work we:

Split the network into full resolution and one-half resolution channels

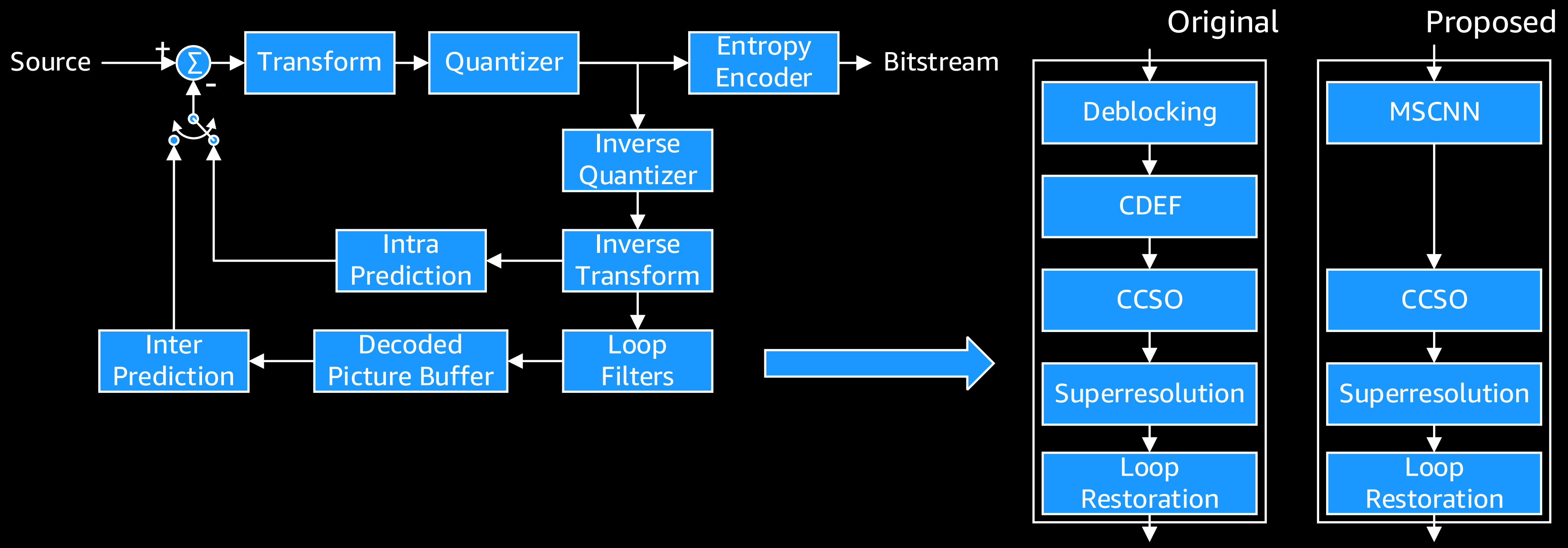
Investigate different approaches for re-combining the full resolution and one-half resolution channels, and

Use 1x1 convolutional layers to manage spatial support

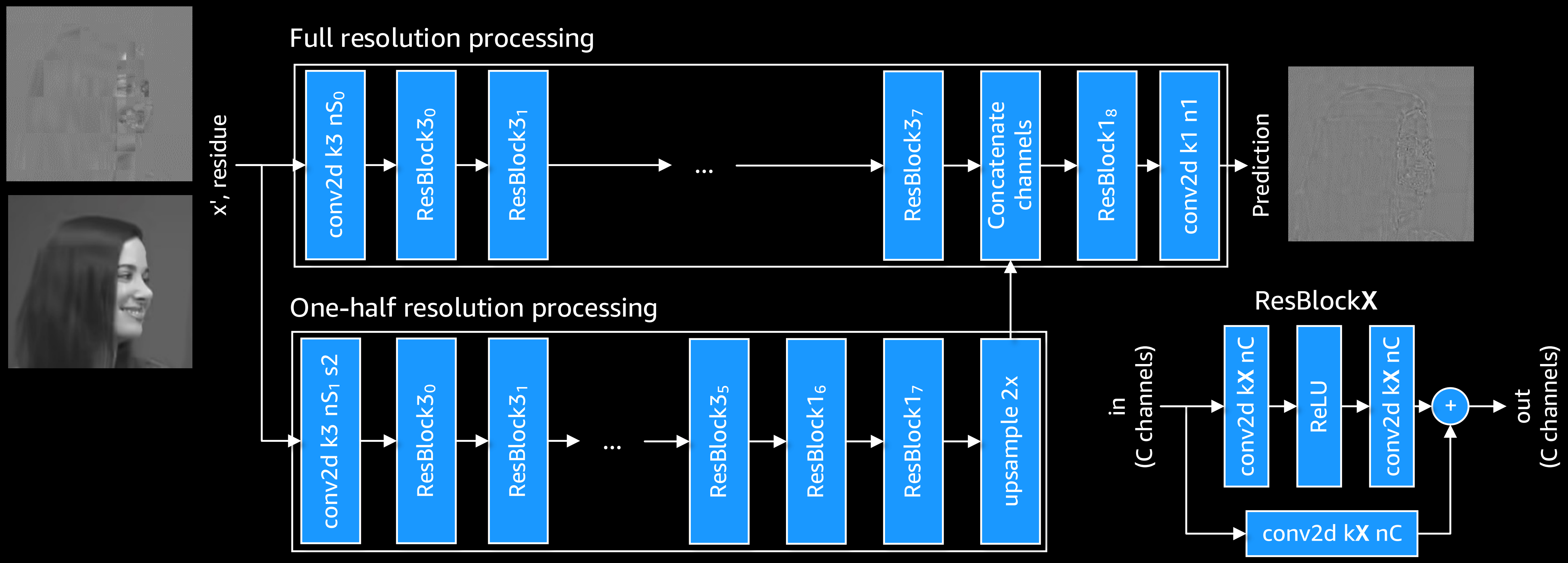
Placement and input

We place MSCCN at the start of the reconstruction loop

Inputs to MSCNN are luma samples and transform residue



Architecture



Design features to improve coding efficiency

The prediction/correction samples values output by MSCNN are scaled using scaling factors 1.0, or 0.75, or 0.50

Application of MSCNN is controlled at block level. Block sizes can be 16x16, 32x32, 64x64 or 128x128

Models are selected based on slice type (intra/inter) and QP, where QP is:

$$QP = (\text{base qindex}) - 24 * (\text{source bit depth} - 8)$$

Design features to improve coding efficiency (contd.)

For intra slices, we use one model for each QP range listed below:

[0…100], [101…124], [125…149], [150…174], [175…200], [200…255]

Similarly for inter slices, we use one model for each QP range listed below:

[0…110], [111…135], [136…160], [161…185], [185…210], [211…255]

Since coding artifacts depend on QP and slice type

prime video

TRAINING AND EVALUATION SETUP

Training Setup

Dataset

- Intra: DIV2K dataset
- Inter: BVI_DVC dataset

Patch Size: 256x256

Batch Size: 1

Model Details:

- 6 models, one for each QP range
- 2 model groups. One group for intra, one group for inter.

Training Setup (contd.)

Learning rate:

- Intra: 10^{-5} for first 90% of epochs, 10^{-6} for remaining epochs
- Inter: 10^{-6} for first 90% of epochs, 10^{-7} for remaining epochs

Model initialization:

- Intra: Random
- Inter: Two pass training. First pass uses intra models for initialization. Second pass uses inter models derived in first pass for initialization.

Training Setup (contd.)

Epoch count:

1760 (for intra), 160 (for inter)

Evaluation dataset:

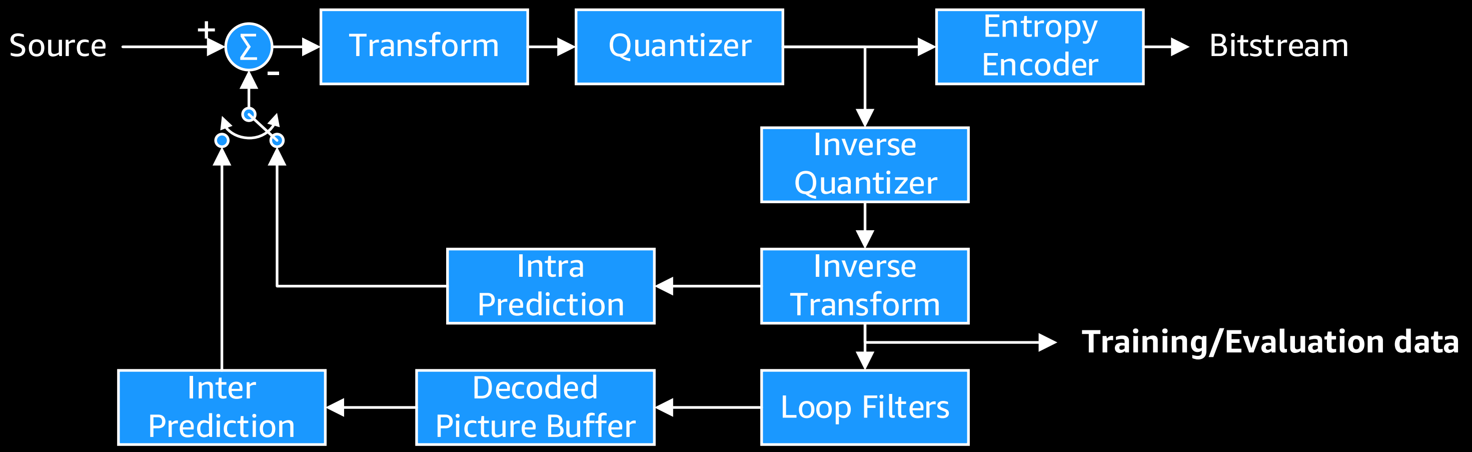
26 pictures from video resolution classes A2 and A3 of AOMedia Common Test Conditions (CTC)

Training loss:

Mean Square Error (MSE)

Training/Evaluation data generation

Training and evaluation data is generated by running AOMedia Common Test Conditions (CTC): All Intra and Random Access configurations



EXPERIMENTAL RESULTS

prime video

Testing

AOMedia Common Test Conditions v3.0

- Intra [30 frames, 6 QPs]
- Random Access [130 frames, 6 QPs]
- Class A sequences

Reference (Anchor): AVM research-v3.0.0

- https://gitlab.com/AOMediaCodec/avm/-/tree/research-v3.0.0

Performance metric

- Bjøntegaard Delta Bitrate

Results

Class	Intra (PSNR BD Rate)				Random Access (PSNR BD Rate)			
	Y	U	V	YUV	Y	U	V	YUV
A1 (4K)	-7.40%	4.13%	4.76%	-6.04%	-6.74%	7.52%	7.67%	-5.13%
A2 (2K)	-7.13%	3.06%	3.53%	-6.16%	-6.89%	4.71%	5.12%	-5.85%
A3 (720p)	-8.90%	3.12%	3.23%	-7.82%	-8.38%	3.81%	4.26%	-7.23%
A4 (360p)	-6.67%	3.79%	4.16%	-5.88%	-7.14%	2.91%	3.27%	-6.35%
A5 (270p)	-6.93%	2.61%	3.84%	-6.09%	-7.72%	0.62%	2.31%	-6.93%
Average	-7.41%	3.34%	3.90%	-6.40%	-7.38%	3.91%	4.53%	-6.30%

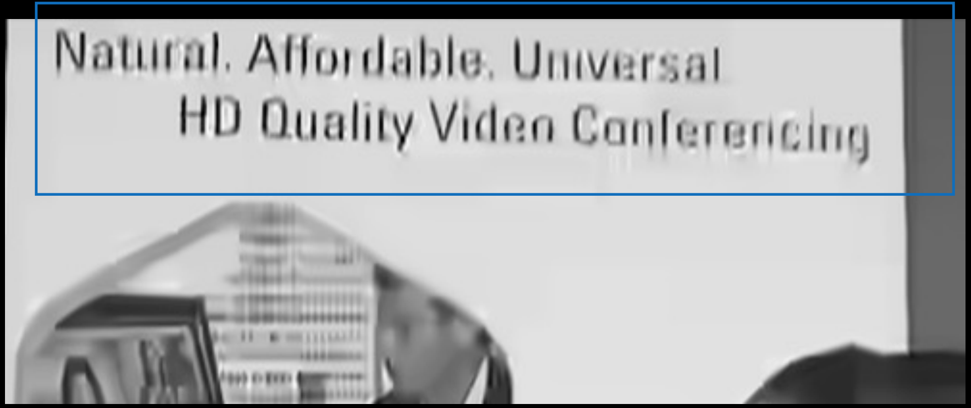
All Intra: -6.4%
Random Access: -6.3%

Results (contd.)

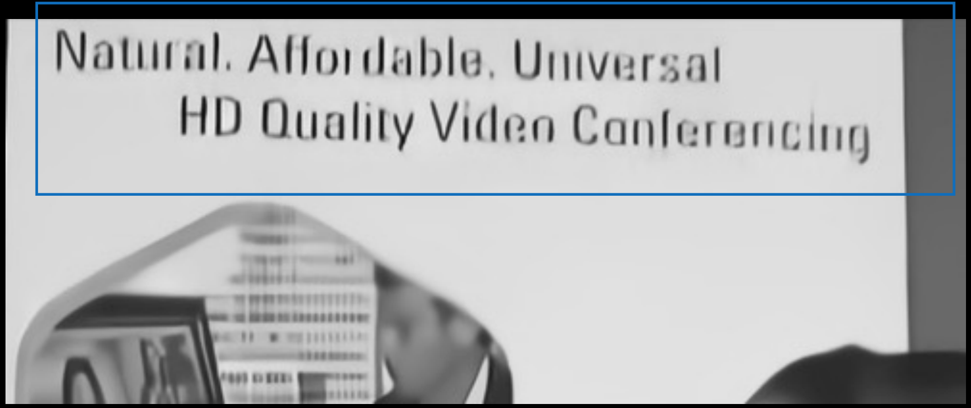
Class	Intra (BD Rate)		Random Access (BD Rate)	
	VMAF Y	nVMAF Y	VMAF Y	nVMAF Y
A1 (4K)	-13.43%	-12.54%	-9.02%	-8.65%
A2 (2K)	-11.95%	-11.12%	-9.62%	-8.71%
A3 (720p)	-12.57%	-11.96%	-10.11%	-9.00%
A4 (360p)	-11.42%	-10.49%	-8.15%	-7.50%
A5 (270p)	-13.98%	-12.76%	-9.91%	-9.11%
Average	-12.67%	-11.77%	-9.36%	-8.59%

The VMAF gains are higher!

Visual comparison of input-output of MSCNN



Input



Output

In example above, MSCNN removes ringing artifacts

ABLATION STUDY

prime video

Aspects studied

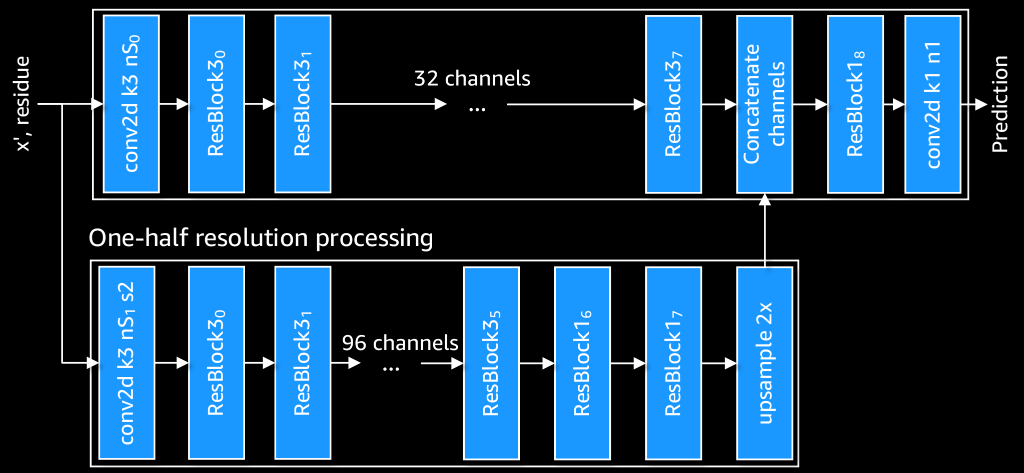
We investigate:

- Impact of multiscale architecture
- Impact of using residual blocks after merging scales
- Impact of using 1x1 convolution in one-half resolution processing path to control spatial support

Metrics:

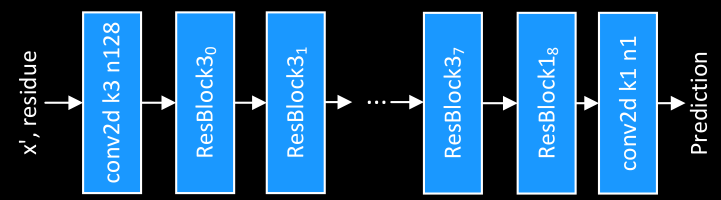
- Coding gain: Intra YUV BD bitrate of class A3, A4, A5 test sequences
- Complexity: MACs per pixel, Spatial Extent

Remove multiscale processing



Complexity-Coding performance of MSCNN

(S_0, S_1)	Parameter count	MACs/pixel	Spatial Extent	YUV BD Rate
(32, 96)	2,073,601	720,752	57x57	-6.60%



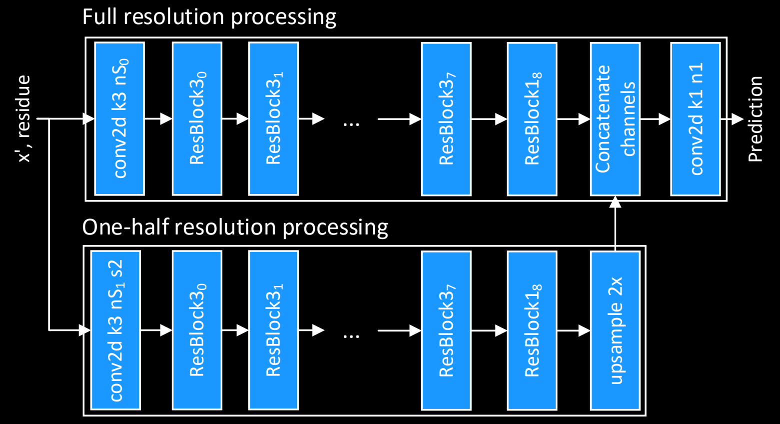
Complexity-Coding performance of full-resolution processing

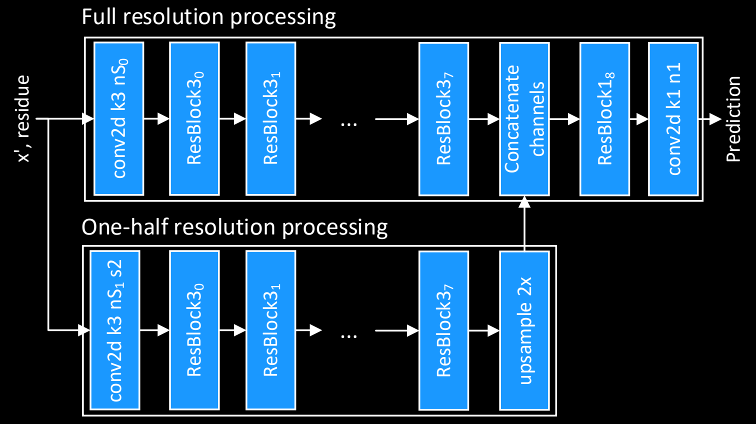
Parameter count	MACs/pixel	Spatial Extent	YUV BD Rate
3,594,113	3,590,528	35×35	-6.83%

Significantly larger MACs/pixel

Removing residual block after merging scales

No residual block after merging scales





Complexity-coding performance when using convolution to merge scales

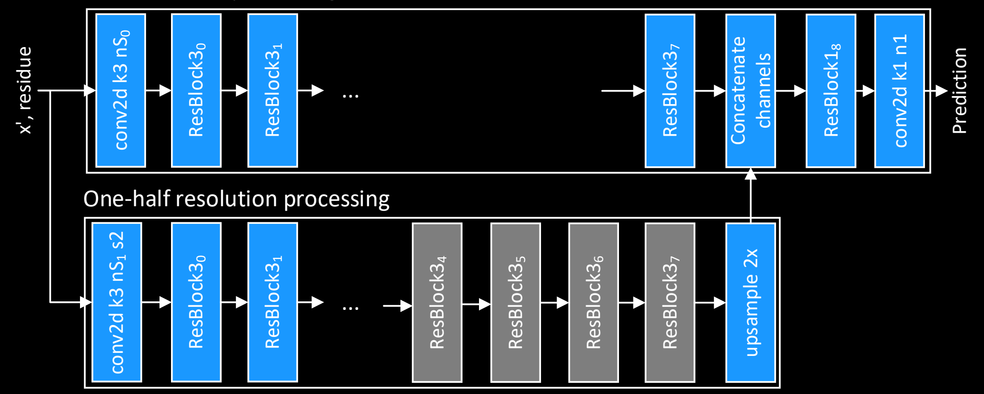
(S_0, S_1)	Parameter count	MACs/pixel	Spatial Extent	YUV BD Rate
(64, 64)	1,800,065	1,123,712	71×71	-6.32%
(32, 96)	2,248,577	731,264	71×71	-5.69%
(16, 112)	2,809,217	745,280	71×71	-5.07%

Complexity-coding efficiency trade-off when using residual block to merge scales

(S_0, S_1)	Parameter count	MACs/pixel	Spatial Extent	YUV BD Rate
(64, 64)	1,824,641	1,157,504	71×71	-6.78%
(32, 96)	2,267,009	770,432	71×71	-6.57%
(16, 112)	2,819,969	784,256	71×71	-6.45%

Higher gains for similar MACs/pixel

Using 1x1 convolution to manage spatial extent



Complexity-coding performance when converting ResBlock3$_a$ to ResBlock3$_b$ into ResBlock1 in one-half resolution processing path

(S_0, S_1)	(a, b)	Parameter count	MACs/Pixel	Spatial Extent	YUV BD Rate
(32, 96)	(6, 7)	1,824,641	659,840	63×63	-6.60%
(32, 96)	(5, 7)	1,603,457	604,544	55×55	-6.50%
(32, 96)	(4, 7)	1,382,273	549,248	47×47	-6.49%

Lower spatial extent, Lower MAC

Summary

Average YUV BD Bitrate (class A sequences):

All Intra: -6.4%

Random Access: -6.3%

In this work we process channels at full and one-half resolution to reduce MAC. 1x1 convolutional layers are used to manage spatial extent. Residual block is used to re-combine the two resolutions for improved coding efficiency

Compared to a similar network that only operates at the high resolution, we observe the multiscale approach reduces complexity by 5.4×

THANK YOU