2017源创会年终盛典

与电子标准院共建开源标准

12月23日　北京万豪酒店

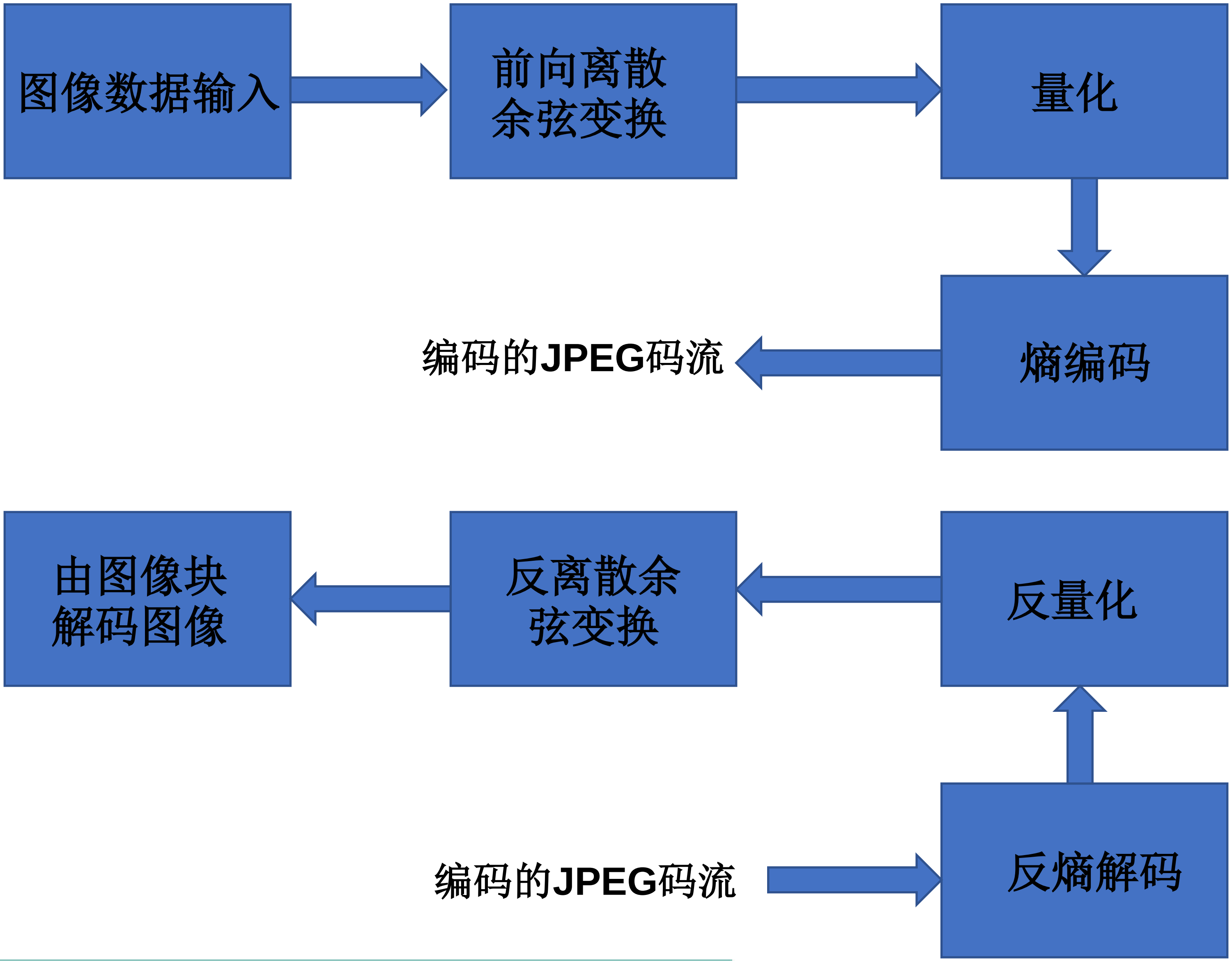
主办方 开源中国 oschina.net

基于深度学习的图像和视频压缩

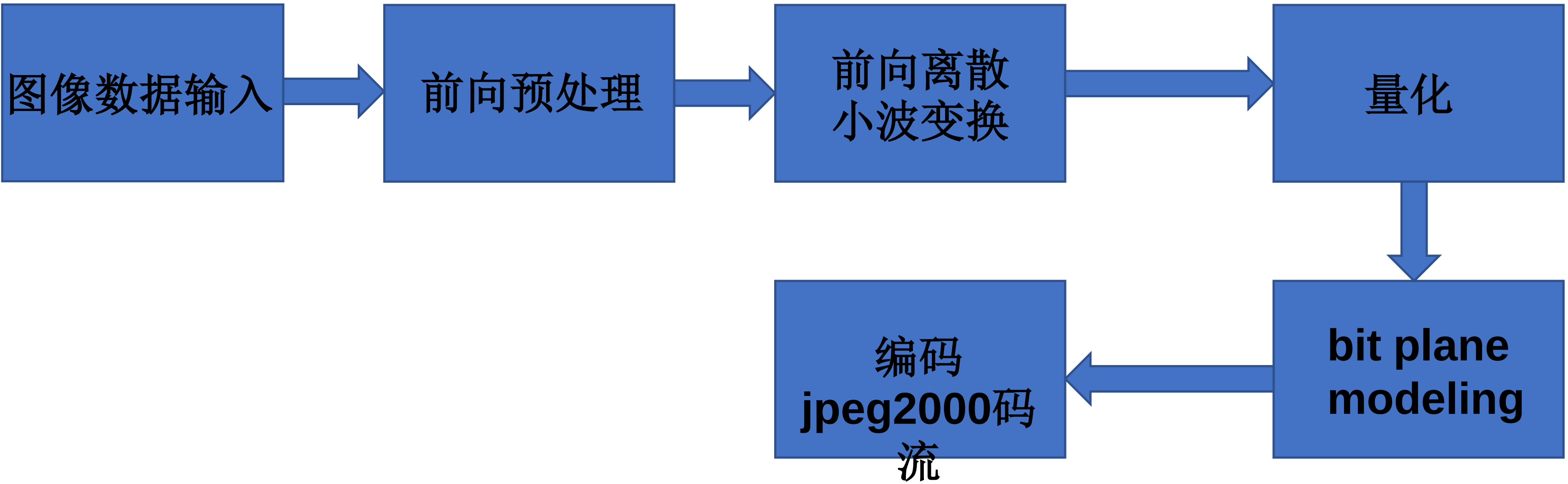
周雷
图鸭科技

目录

◆ 图像压缩-JPEG


图像数据输入
前向离散余弦变换
量化
熵编码
编码的JPEG码流
由图像块解码图像
反离散余弦变换
反量化
编码的JPEG码流
反熵解码

◆ 图像压缩-JPEG2000

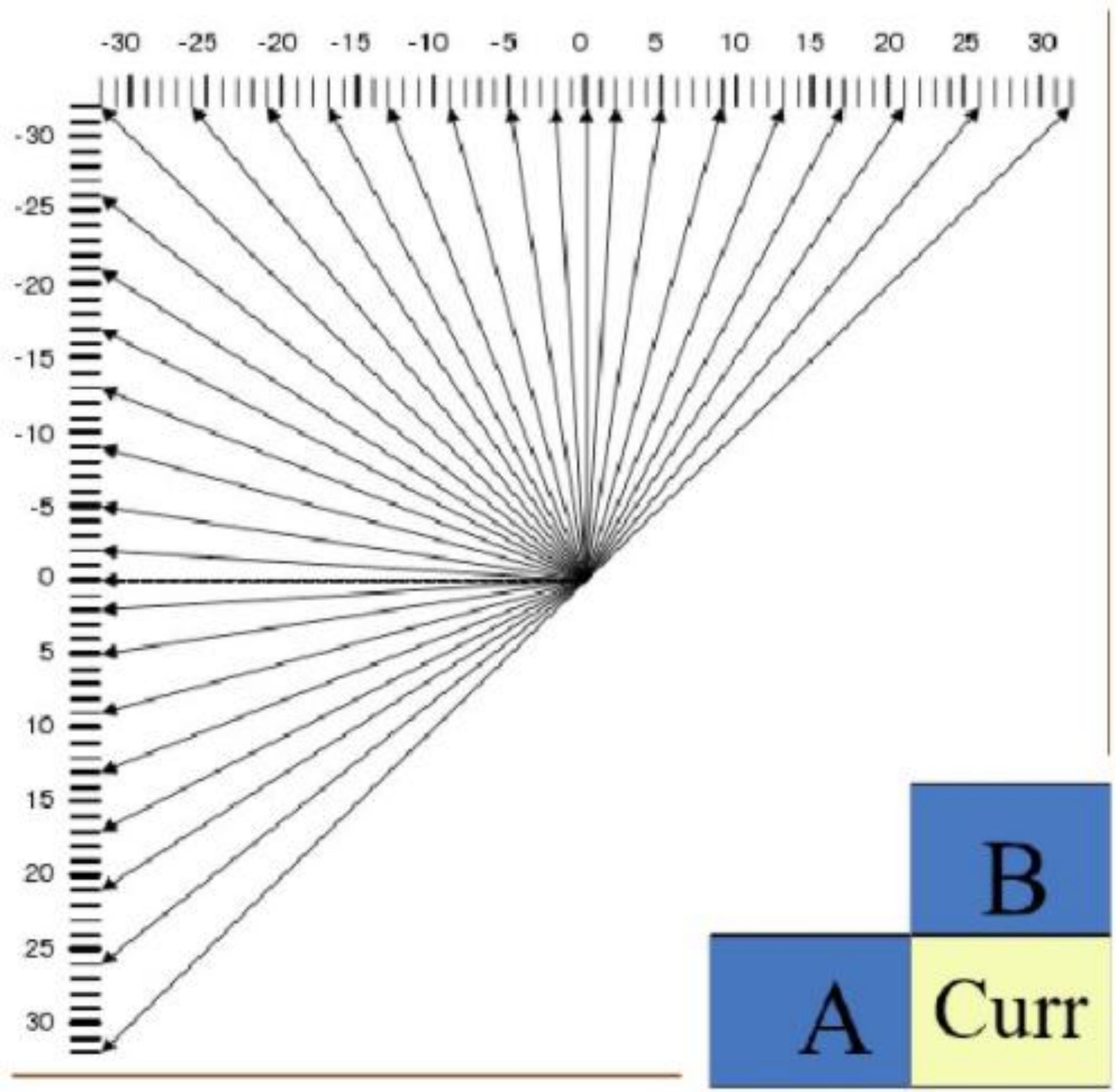


◆ 图像压缩-BPG

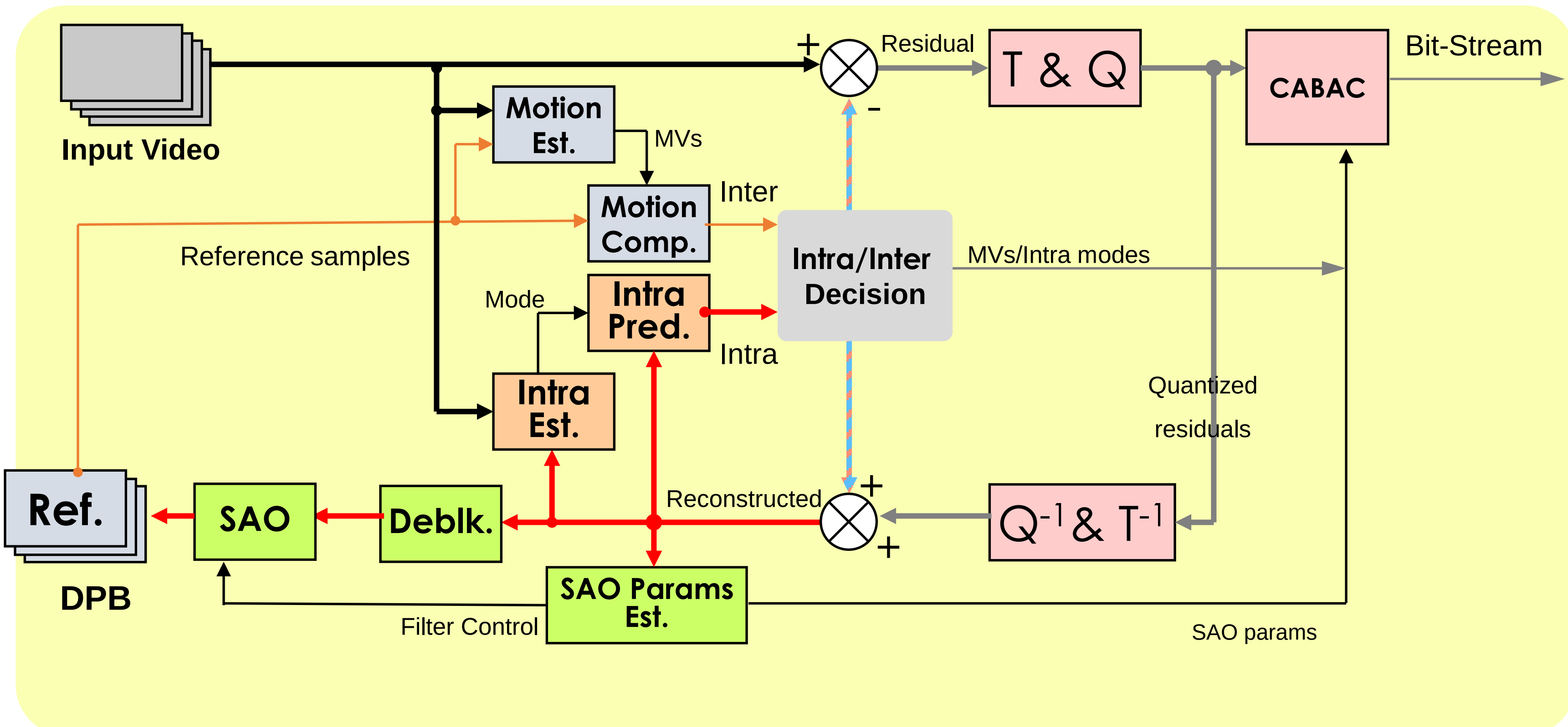
利用像素空间相关性消除空域冗余

多角度预测模式，进行图像内预测

对预测残差进行DCT编码



◆ 视频压缩-HEVC


Input Video
+
Residual
T & Q
CABAC
Bit-Stream
-
Motion Est.
MVs
Motion Comp.
Inter
Intra/Inter Decision
MVs/Intra modes
Reference samples
Mode
Intra Pred.
Intra
Intra Est.
Quantized
residuals
Ref.
DPB
SAO
Deblk.
Reconstructed
+
+
Q^{-1} & T^{-1}
SAO Params Est.
Filter Control
SAO params

◆ 深度学习与图像视频压缩

如何用深度学习来设计图像或视频压缩算法？

深度学习在如人脸等视觉领域取得飞速发展，能否对图像视频压缩领域进行更新？

深度学习框架有哪些独特的优势？

典型的深度学习图像压缩框架介绍

$\mathbb{R}^{128\times128\times3}$

原始图片

CNN Encoder

编码

$c \in \Re^{8\times8\times256}$

Quanti

量化

q

Entropy Coding

熵编码

Bits

$q^{'} \in \Re^{8\times8\times256}$

Dequant

反量化

CNN Decoder

解码

原图

重建误差

码字估计

ratio-distoration cost

码字计算： $rate = \dfrac{\boxed{8\times8\times256}\times R}{128\times128} = R$ (bit/pixel)

与网络相关参数

每个数据单元平均耗费的比特位数

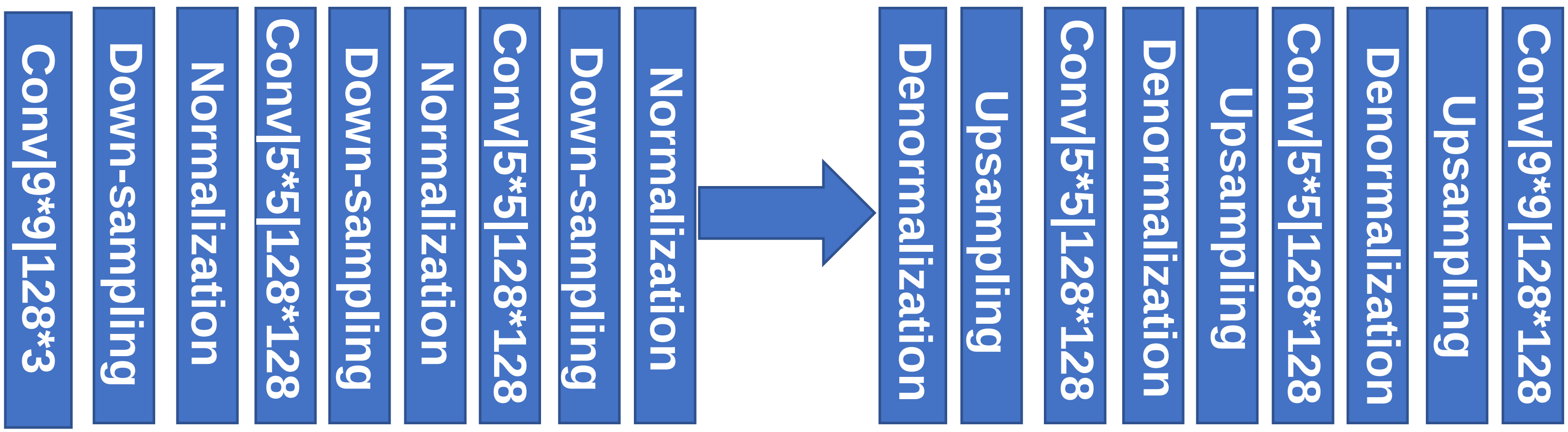
与码字分布和熵编码有关

To generate a binary representation for an image by quantizing

完整的深度压缩网络所包括的模块

1 编码和解码网络
2 量化和反量化
3 重建误差
4 熵编码和码字估计
5 ratio-distoration cost

编码和解码网络结构



- 编码网络结构可由卷积、池化等模块组成，如 AlexNet etc.
- 解码网络结构可由上采样，卷积，反卷积等模块组成

量化与反量化

前向过程中，浮点数可用整数加随机噪声来表示

$$\{y\} \approx \lfloor y \rfloor + \varepsilon, \quad \varepsilon \in \{0, 1\}, \quad P(\varepsilon = 1) = y - \lfloor y \rfloor,$$

在梯度反向传播过程中, 梯度可用期望的梯度来表示

$$\frac{d}{dy}\{y\} := \frac{d}{dy}\mathbb{E}\left[\{y\}\right] = \frac{d}{dy}y = 1.$$

LOSSY IMAGE COMPRESSION WITH COMPRESSIVE AUTOENCODERS, ICLR 2017

重建误差

- 重建误差 - MSE.

To measure the average of the squares of the errors between original images and reconstructed images.

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(\hat{Y_i} - Y_i)^2$$

熵编码和码字估计

约束码字和约束编码特征的分布，从而产生更小的码字

现有的方法:

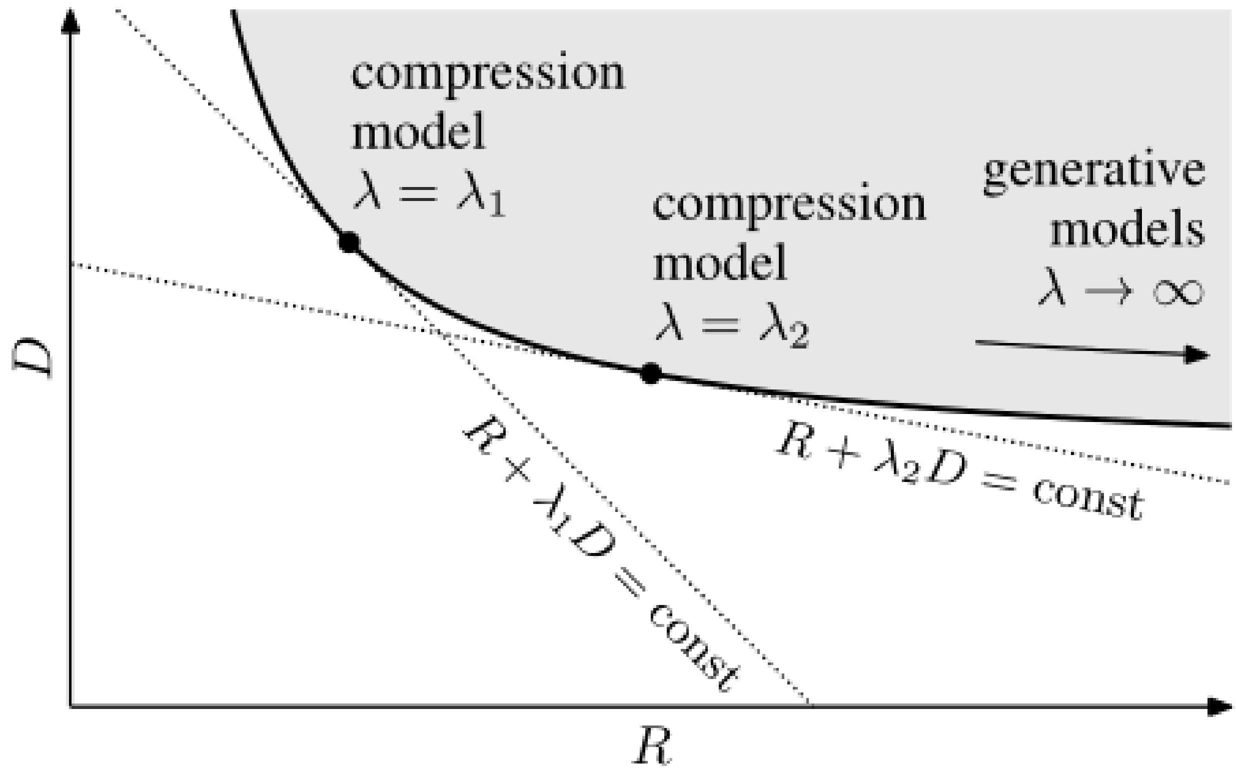
BinaryRNN for Entropy Coding--
Full Resolution Image Compression with Recurrent Neural networks.CVPR17

Bitplane Decomposition for Adaptive Arithmetic Coding--
Real-Time Adaptive Image Compression, ICML 2017

Gaussian Model for Entropy Rate Estimation--
Lossy Image Compression With Compressive Autoencoders, ICLR 2017

End to End Optimized Image Compression, ICLR 2017

网络的优化：RD-COST



$$L[g_a, g_s, P_q] = -\,\mathbb{E}\left[\log_2 P_q\right] + \lambda\,\mathbb{E}\left[d(z, \hat{z})\right]$$

码字估计

重建误差

如何进行测评

数据库: Kodak Photo CD dataset

- 24 张768×512 PNG 图片

评价指标

- MS-SSIM （主观指标）
- PSNR （客观指标）

比较

- JPEG
- JPEG2K
- BPG
- 不同网络结构
  - LSTM, Associative LSTM, GRU
- 不同的重建框架
- One-shot, additive, residual scaling

典型方法介绍

两个主要的发展方向

- 端到端模式 (auto-encoders)
  - RNN (ICLR2016_Google, CVPR2017_Google, ICIP2017_Google)
  - CNN (ICLR2017_NYU, ICLR2017_Twitter, CSVT_HIT)
  - GAN (ICML2017_waveone, MIT_2017)
- 优化现在的视频编码器

  Intra CU mode decision

  Down-samping Coding

  in-loop filter & Post-processing

ICLR2016_Google

- RNN编码层
- 二进制量化层
- RNN解码层

$$b_t = B(E_t(r_{t-1})), \quad \hat{x}_t = D_t(b_t) + \gamma \hat{x}_{t-1}, \quad r_t = x - \hat{x}_t, \quad r_0 = x, \quad \hat{x}_0 = 0$$

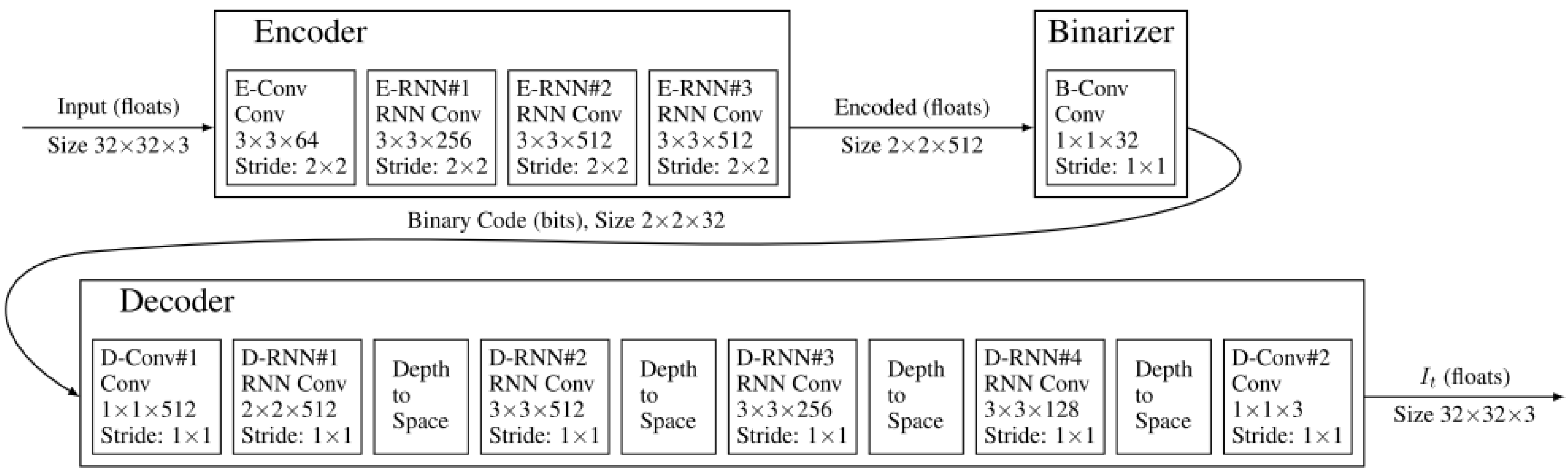
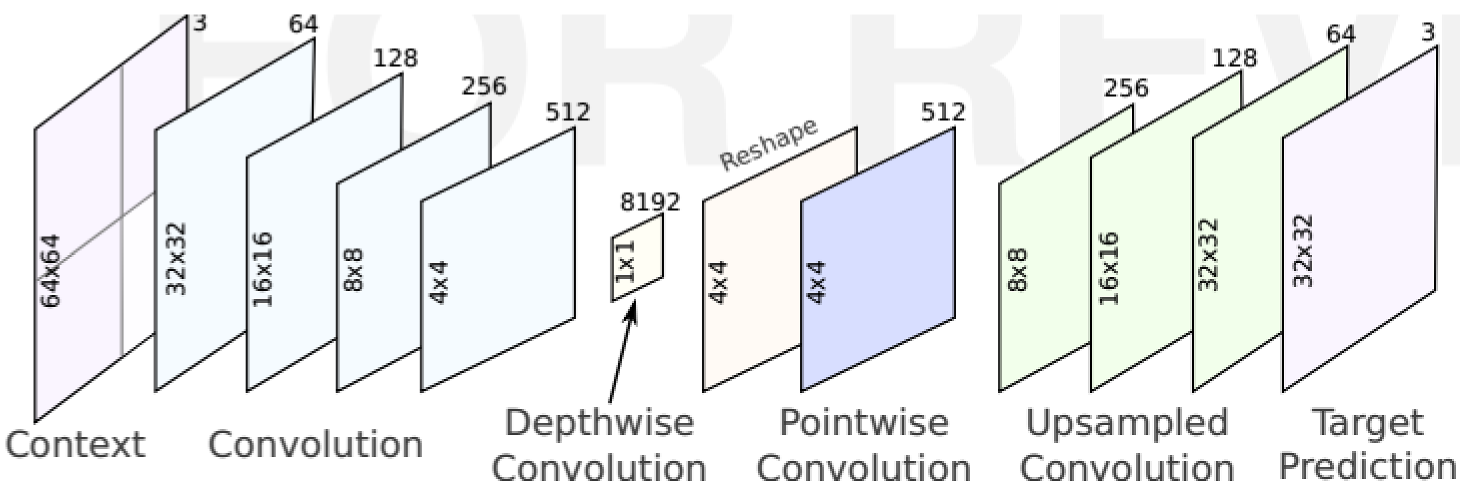


Figure 1: A single iteration of our shared RNN architecture.

ICIP2017_Google

RNN with intra prediction



两个主要的发展方向

- 端到端模式 (auto-encoders)
  - RNN (ICLR2016_Google, CVPR2017_Google, ICIP2017_Google)
  - CNN (ICLR2017_NYU, ICLR2017_Twitter, CSVT_HIT)
  - GAN (ICML2017_waveone, MIT_2017)
- 优化现在的视频编码器

  Intra CU mode decision

  Down-samping Coding

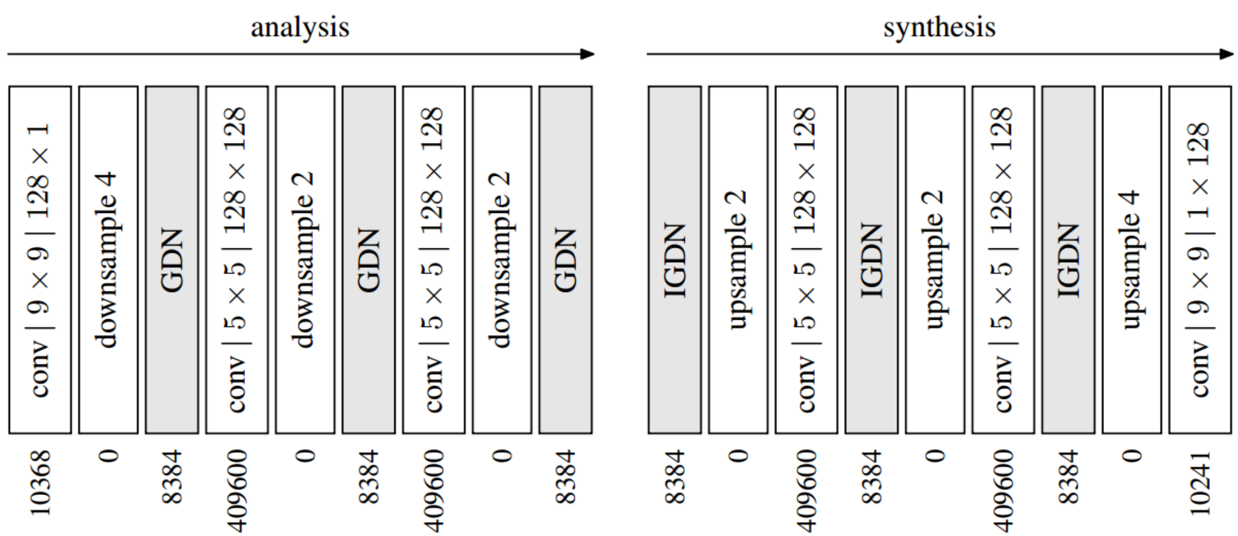
  in-loop filter & Post-processing

ICLR2017_NYU

- Generalized divisive normalization (GDN)

$$u_i^{(k+1)}(m,n) = \frac{w_i^{(k)}(m,n)}{\left(\beta_{k,i} + \sum_j \gamma_{k,ij}\left(w_j^{(k)}(m,n)\right)^2\right)^{\frac{1}{2}}}.$$

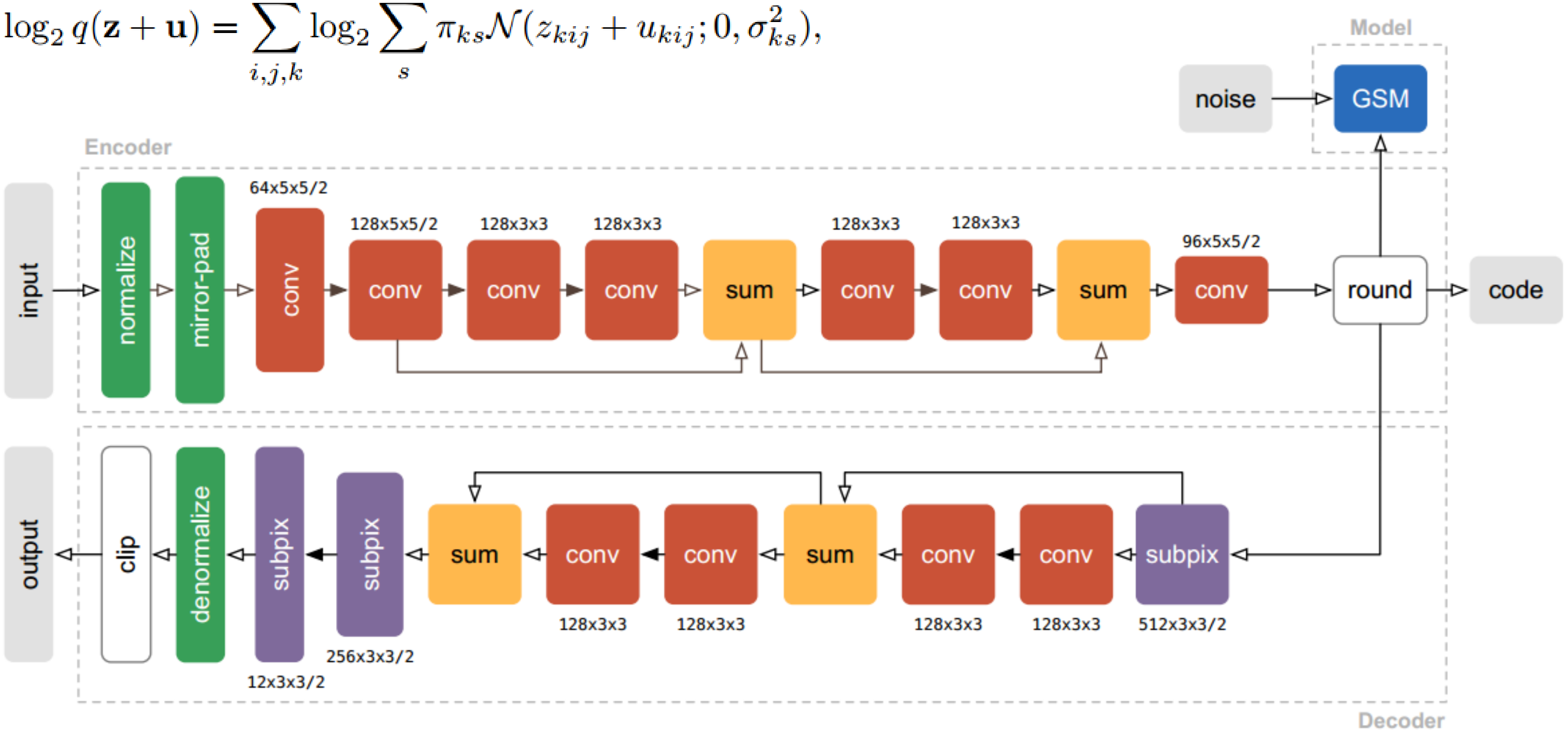
- 网络结构和码字估计



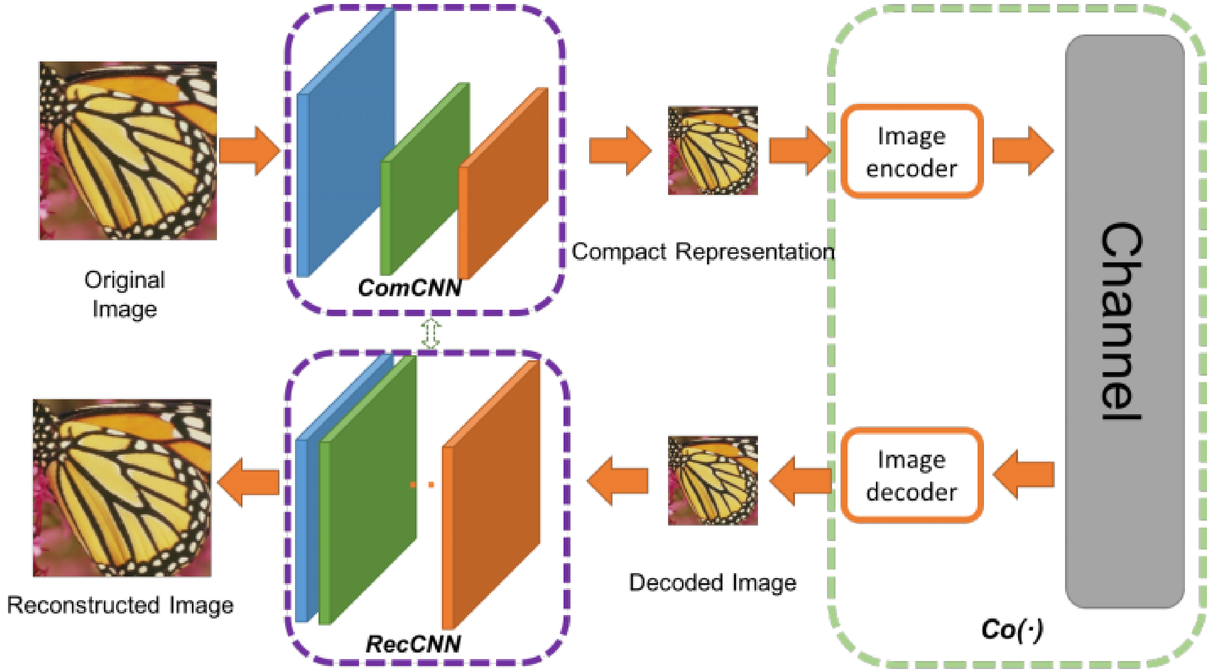
ICLR2017_Twitter

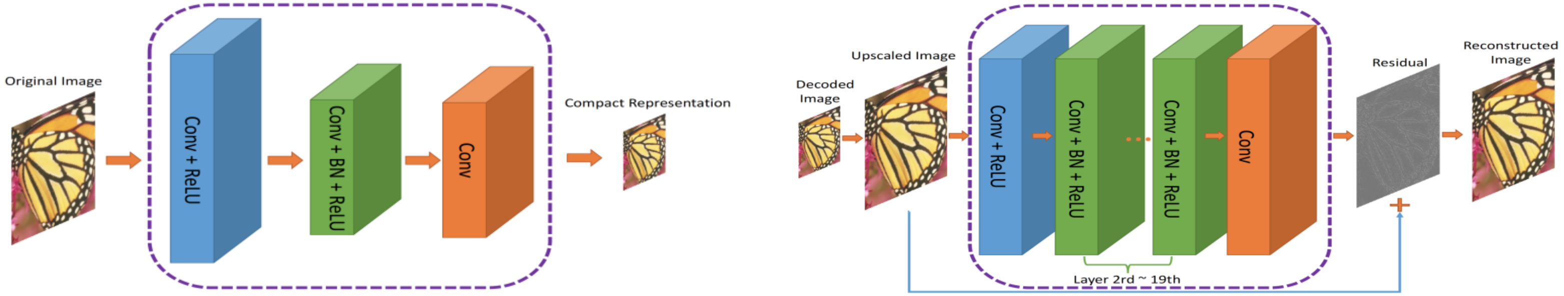
Rate estimation by Gaussian Scale Mixtures

$$\log_2 q(\mathbf{z}+\mathbf{u}) = \sum_{i,j,k} \log_2 \sum_s \pi_{ks} \mathcal{N}(z_{kij} + u_{kij}; 0, \sigma_{ks}^2),$$



CSVT2017_HIT





两个主要的发展方向

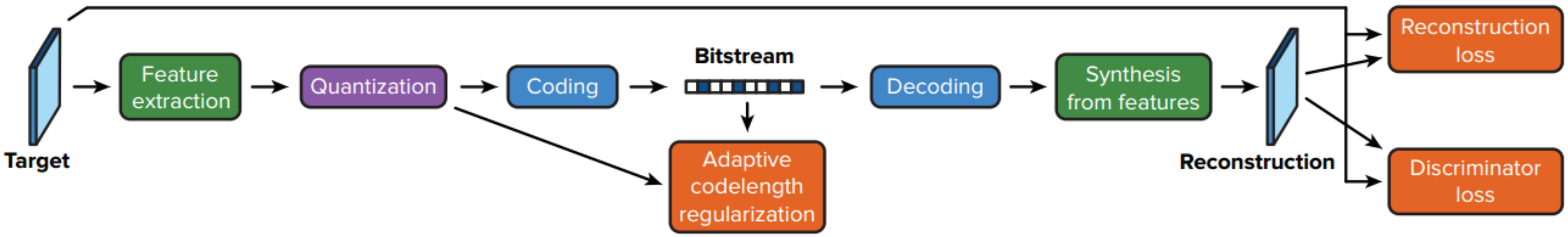
- 端到端模式 (auto-encoders)
  - RNN (ICLR2016_Google, CVPR2017_Google, ICIP2017_Google)
  - CNN (ICLR2017_NYU, ICLR2017_Twitter, CSVT_HIT)
  - GAN (ICML2017_waveone, MIT_2017)
- 优化现在的视频编码器

  Intra CU mode decision

  Down-samping Coding

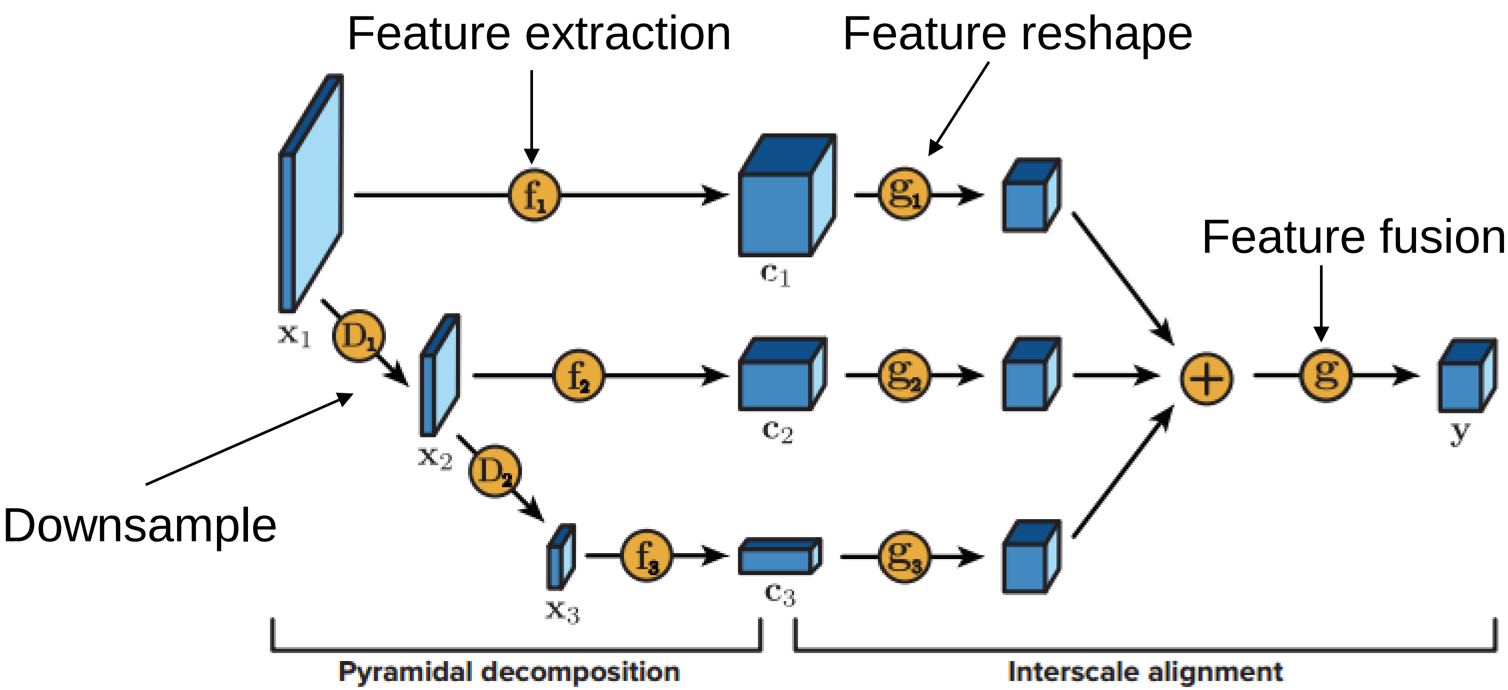
  in-loop filter & Post-processing

ICML2017_waveone



- 编码网络
  - Multiscale analysis
- 量化和熵编码
  - Adaptive arithmetic coding and adaptive codelength regularization
- 重建误差(Discriminator loss)

编码网络



Feature extraction
Feature reshape
Feature fusion
Downsample
f_1
f_2
f_3
g_1
g_2
g_3
g
D_1
D_2
x_1
x_2
x_3
c_1
c_2
c_3
y
Pyramidal decomposition
Interscale alignment

量化和熵编码

- 量化:

$$\hat{y}_{chw} := \textsc{Quantize}_B(y_{chw}) = \frac{1}{2^{B-1}} \left\lceil 2^{B-1} y_{chw} \right\rceil$$

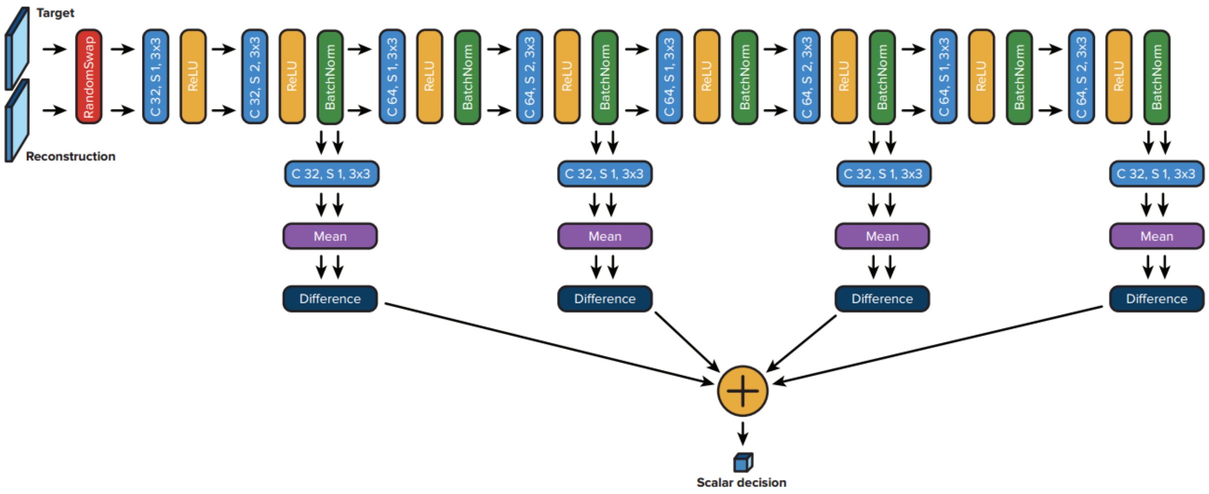
- 适应性算术编码 (AAC)
  - Train context model to estimate probability of each bin
- 适应性码字长度约束 (ACR)
  - Obtain target codelength while training

$$\mathscr{P}(\hat{\mathbf{y}}) = \frac{\alpha_t}{CHW} \sum_{chw} \left\{ \log_2 |\hat{y}_{chw}| + \sum_{(x,y)\in S} \log_2 \left|\hat{y}_{chw} - \hat{y}_{c(h-y)(w-x)}\right| \right\},$$

$\hat{\mathbf{y}}_c \in \mathbb{R}^{H\times W}$ Bitplane

重建误差(Discriminator loss)

- Discriminator:
  - Input: target and reconstruction
  - Multiscale discriminator
- Adversarial training
  - Alternatively train coding engine and discriminator



两个主要的发展方向

- 端到端模式 (auto-encoders)
  - RNN (ICLR2016_Google, CVPR2017_Google, ICIP2017_Google)
  - CNN (ICLR2017_NYU, ICLR2017_Twitter, CSVT_HIT)
  - GAN (ICML2017_waveone, MIT_2017)
- CNN与现有的视频编码器结合

  Intra CU mode decision

  Down-samping Coding

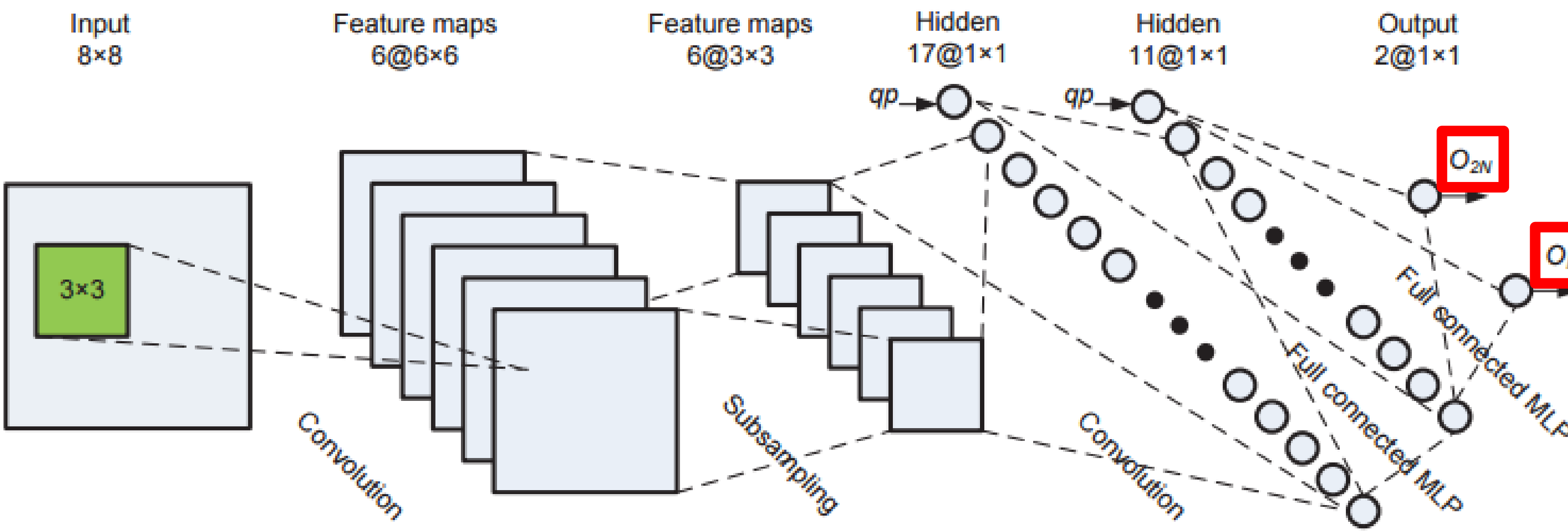
  in-loop filter & Post-processing

编码单元模式选择(Intra CU mode decision)

- CNN oriented fast HEVC intra CU mode decision
  - Contributions:
  - 1. Using CNN to analyze the textures of CU
  - 2. Reduce the maximum number of CU modes
  - 3. Introduce QP into CNN architecture design

Liu Z, Yu X, Chen S, et al. CNN oriented fast HEVC intra CU mode decision. ISCAS 2016: 2270-2273.

编码单元模式选择(Intra CU mode decision)



(b) Architecture of Convolutional Neural Network for CU Mode Decision

- Objective: learning to classify 2Nx2N or NxN
  - The output of two nodes are RD-Cost

编码单元模式选择(Intra CU mode decision)

- **时间和码字节省效果测评**
  - 63% time save with 2.7% loss in BDBR

Table III: Performance Comparison between Proposed Solution and Existing Algorithms

Algorithm	$\Delta T_{CMD}[\%]$	$\Delta T_{PMD}[\%]$	BDBR[%]	$\Delta T[\%]$	VLSI
[3]	50-α	α	0.7	50	No
[5]†	26	45	1.0	60	No
[6]	52	0	0.8	52	No
[7]†	52	5	5.1	57	Yes
[8]	62	0	4.5	62	Yes
Proposed	63	0	2.7	63	Yes

† indicates that class F sequences were not tested.

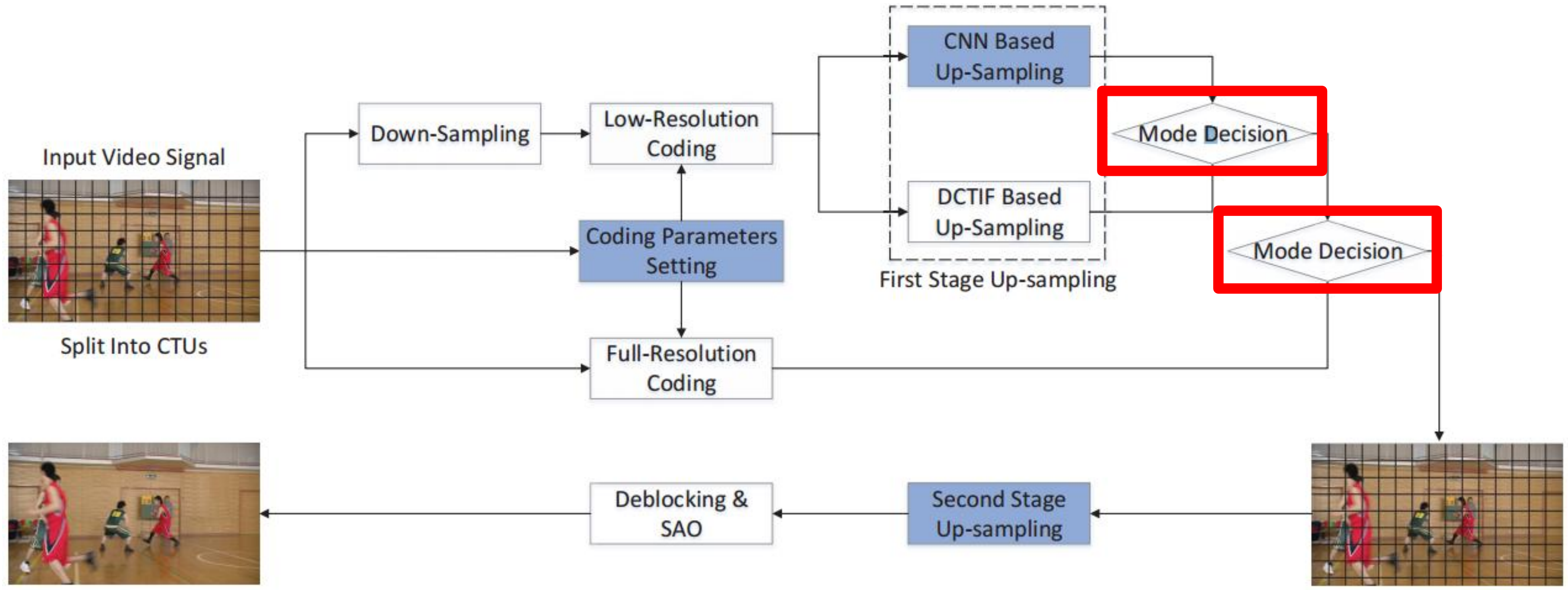
Downsampling-coding (下采样编码)

- 1. CTU (编码树单元)
- 2. Frame (视频帧)

1. Li Y, Liu D, Li H, Li L, Wu F. Convolutional Neural Network-Based Block Up-sampling for Intra Frame Coding.
2. Jia C, Zhang X, Zhang J, et al. Deep Convolutional Network based Image Quality Enhancement for Low Bit Rate Image Compression.

Downsampling-coding(下采样编码)

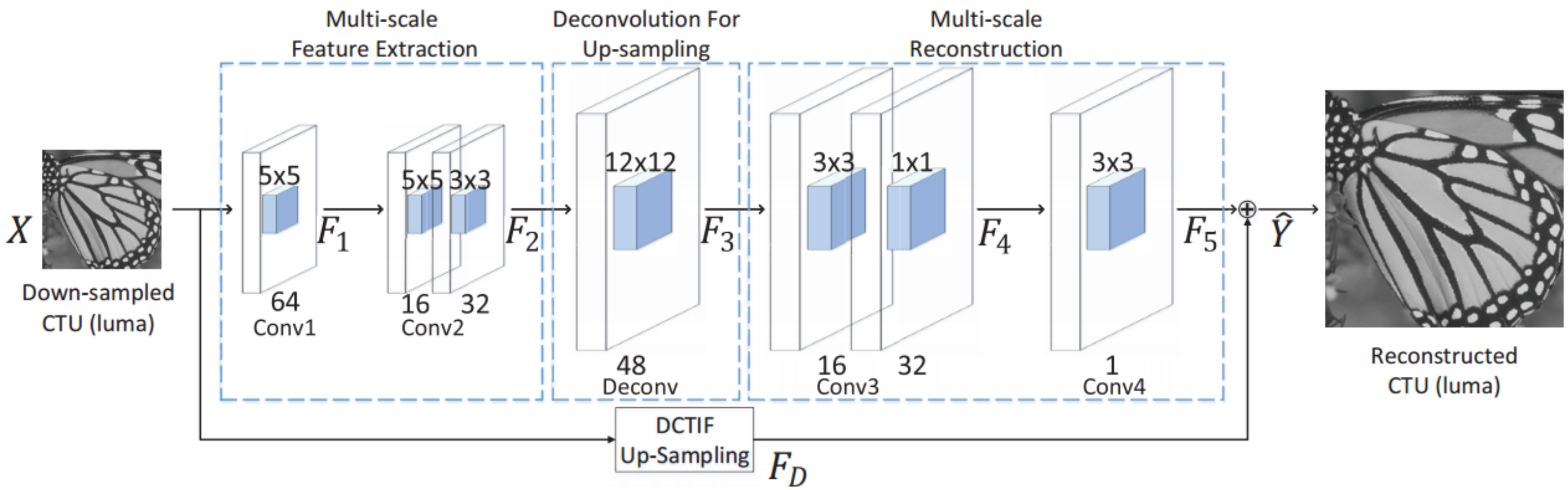
- CTU 层上的操作
  - Two steps RDO
    - 1. Down-sample coding / Full resolution coding
    - 2. CNN up-sampling / DCTIF sampling



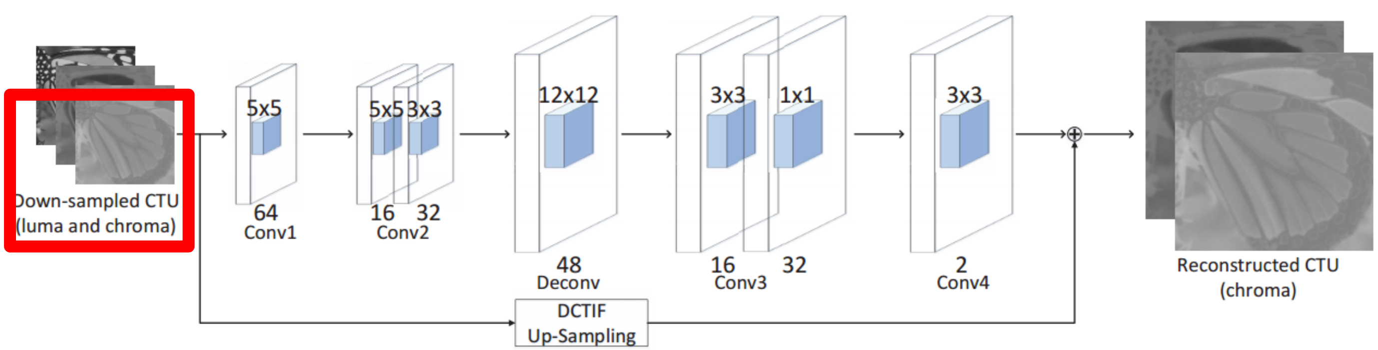
视频帧层上的操作

Downsampling-coding(下采样编码)

- 对亮度通道
  - Input: low resolution patch, output: high resolution



- 对色度通道: downsampled luma also input



Downsampling-coding (下采样编码)

- Test Condition
  - Qp: 32, 37, 42, 47

Class	Sequence	BD-Rate (Anchored on HEVC)				BD-Rate (Anchored on HEVC+DCTIF)			
		Y	U	V	Y SSIM	Y	U	V	Y SSIM
Class A	Traffic	–10.1%	–3.5%	6.0%	–12.9%	–8.0%	–13.2%	–2.6%	–7.9%
	PeopleOnStreet	–9.7%	–14.8%	–14.5%	–12.9%	–8.5%	–20.4%	–18.5%	–9.7%
	Nebuta	–2.0%	–22.0%	3.1%	–4.4%	–1.7%	–22.5%	1.6%	–3.6%
	SteamLocomotive	–1.7%	–27.7%	–25.4%	–6.1%	–1.2%	–34.2%	-25.6%	–2.8%
Class B	Kimono	–7.7%	–5.5%	18.8%	–9.6%	–3.4%	–25.9%	–4.3%	–3.4%
	ParkScene	–7.1%	–14.4%	–2.3%	–11.3%	–5.0%	–25.2%	–14.6%	–6.6%
	Cactus	–6.6%	–2.5%	8.3%	–10.0%	–5.0%	–6.5%	0.9%	–6.7%
	BQTerrace	–3.7%	–7.6%	–9.1%	–9.6%	–3.1%	–8.2%	–7.1%	–6.5%
	BasketballDrive	–6.1%	–1.2%	3.2%	–10.8%	–3.4%	–5.8%	–2.5%	–3.8%
Class C	BasketballDrill	–4.9%	4.5%	8.1%	–7.9%	–4.0%	4.9%	2.1%	–6.6%
	BQMall	–2.9%	–7.2%	–7.2%	–6.2%	–2.3%	–10.6%	–9.1%	–5.3%
	PartyScene	–1.0%	–5.1%	–1.6%	–4.0%	–1.0%	–5.5%	–3.2%	–3.6%
	RaceHorsesC	–6.7%	4.6%	7.5%	–10.7%	–6.0%	1.9%	3.9%	–8.6%
Class D	BasketballPass	–2.0%	–3.7%	9.2%	–4.3%	–2.3%	–7.5%	12.3%	–4.4%
	BQSquare	–0.9%	–0.6%	–21.1%	–1.4%	–0.5%	1.7%	-16.7%	–1.2%
	BlowingBubbles	–3.2%	3.1%	–8.0%	–5.3%	–1.7%	0.5%	-9.6%	–3.8%
	RaceHorses	–9.9%	7.5%	6.4%	–12.6%	–9.6%	5.0%	6.6%	–11.1%
Class E	FourPeople	–7.2%	–10.5%	–11.0%	–11.0%	–7.2%	–14.7%	–14.5%	–9.5%
	Johnny	–9.0%	–3.2%	–3.2%	–11.1%	–7.1%	–6.0%	–8.3%	–5.6%
	KristenAndSara	–6.8%	–11.2%	–11.1%	–13.0%	–5.3%	–8.4%	–10.6%	–8.2%
Class UHD	Fountains	–4.0%	–12.9%	–11.2%	–7.4%	–2.0%	–16.1%	–9.2%	–2.0%
	Runners	–11.2%	22.8%	–0.1%	–12.4%	–7.0%	0.9%	–13.7%	–6.0%
	Rushhour	–8.5%	4.4%	1.8%	–10.3%	–3.2%	–9.2%	–9.5%	–3.0%
	TrafficFlow	–12.7%	–11.7%	–5.8%	–12.7%	–6.9%	–17.3%	–11.9%	–5.6%
	CampfireParty	–8.4%	–10.8%	–0.8%	–9.5%	–6.5%	–10.8%	–5.0%	–6.4%
Average of Classes A-E		–5.5%	–6.0%	–2.2%	–8.8%	–4.3%	–10.0%	–6.0%	–5.9%
Average of Class UHD		–9.0%	–1.6%	–3.2%	–10.5%	–5.1%	–10.5%	–9.9%	–4.6%

In-loop filter & Post Processing

- 视频帧环路滤波和后处理

1. In-loop

Park W S, Kim M. CNN-based in-loop filtering for coding efficiency improvement. IEEE Image, Video, and Multidimensional Signal Processing Workshop (IVMSP) 2016: 1-5.

2. Post Processing

Dai Y, Liu D, Wu F. A Convolutional Neural Network Approach for Post-Processing in HEVC Intra Coding. MMM 2017: 28-39.

In-loop filter

- 视频帧图像滤波
  - In-loop

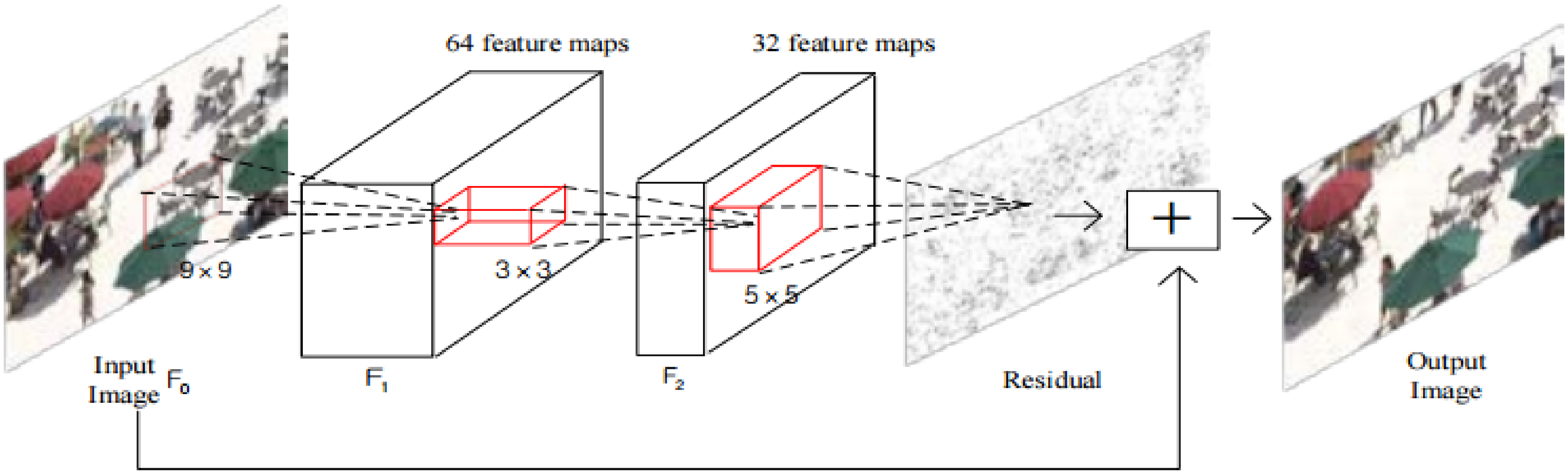
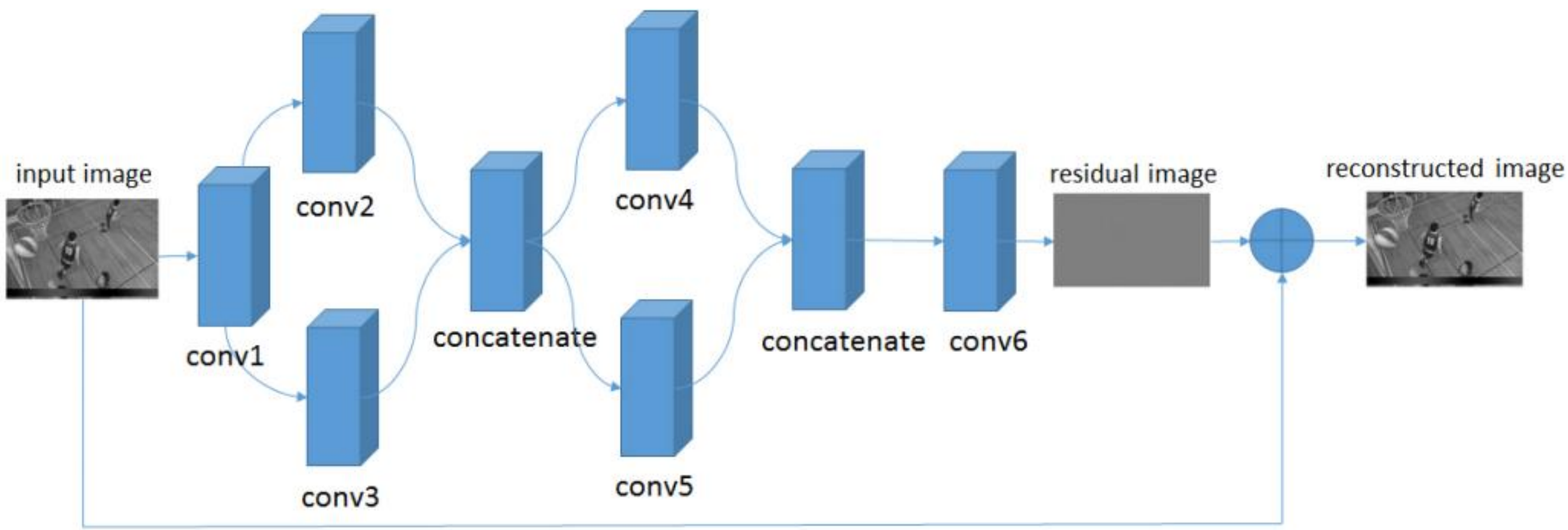


Table 1. Performance of our proposed IFCNN in comparison with SAO in terms of BD rates (BDBR).

Sizes	Seq.	All Intra	LDP-Case I	LDP-Case II	RA-Case I	RA-Case II
		BDBR (%)	BDBR (%)	BDBR (%)	BDBR (%)	BDBR (%)
832×480	BD	-10.1	-5.3	-3.0	-6.0	-6.7
	BQM	-3.7	-3.0	-2.4	-2.4	-2.9
	PS	-2.7	-2.0	-1.2	0.0	-1.1
	BDT	-7.6	-3.5	-2.4	-4.3	-4.9
416×240	BP	-3.3	-2.8	-1.5	-0.6	-1.1
	BQS	-2.4	-3.3	-2.9	1.4	-0.8
	B	-3.4	-2.3	-2.6	0.0	-1.4
	RH	-4.9	-0.4	0.6	-1.2	-1.6
	Avg.	-4.8	-2.8	-1.9	-1.6	-2.6

Post-processing

- 视频帧图像滤波
  - Post processing, All Intra
  - QP: 22, 27, 32, 37

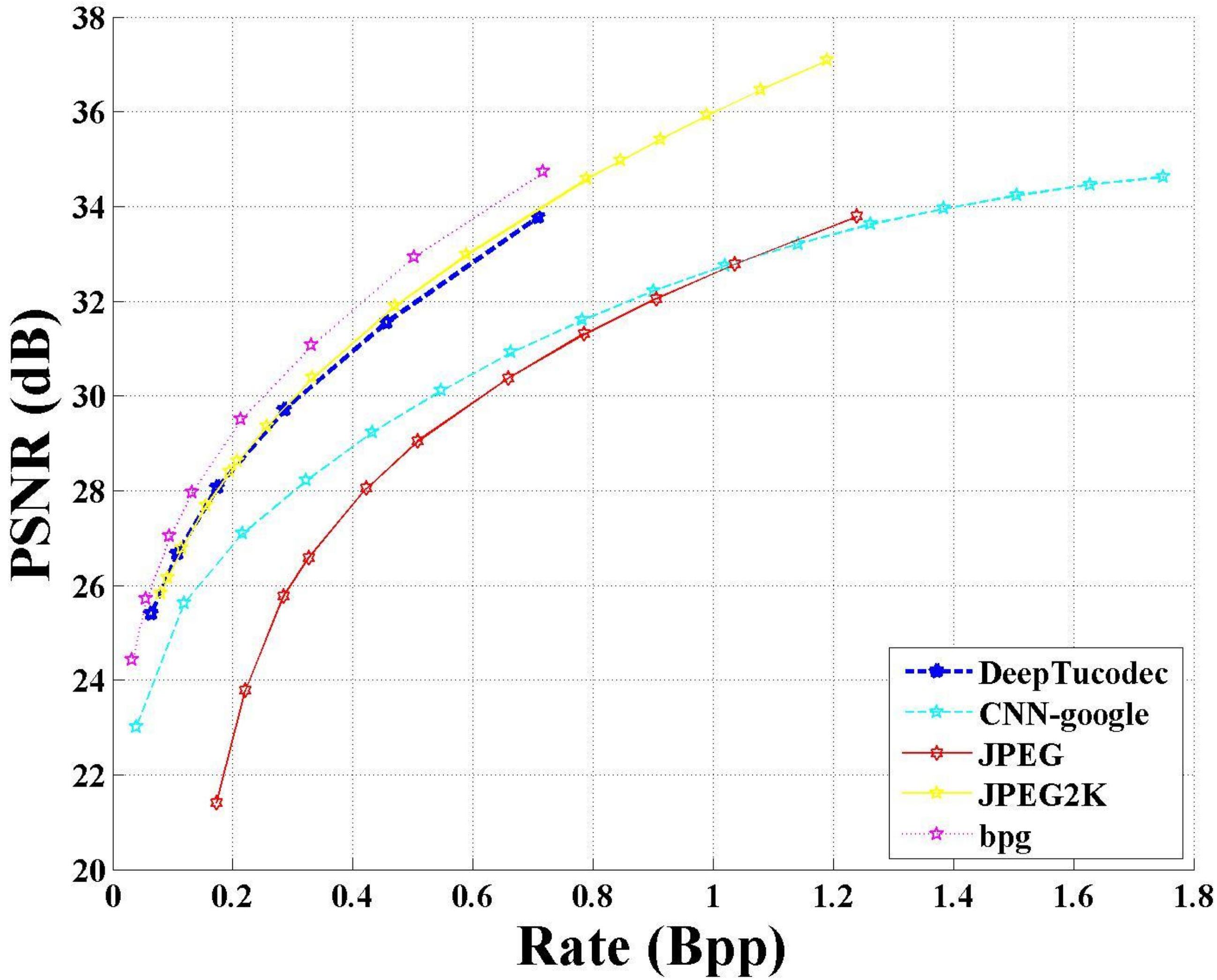


VDSR	Class A	-2.8	-3.2	-3.1
	Class B	-2.7	-2.7	-3.3
	Class C	-4.1	-4.8	-5.7
	Class D	-4.4	-5.6	-7.3
	Class E	-5.7	-5.7	-6.1
	Overall	**-3.8**	**-4.3**	**-4.9**

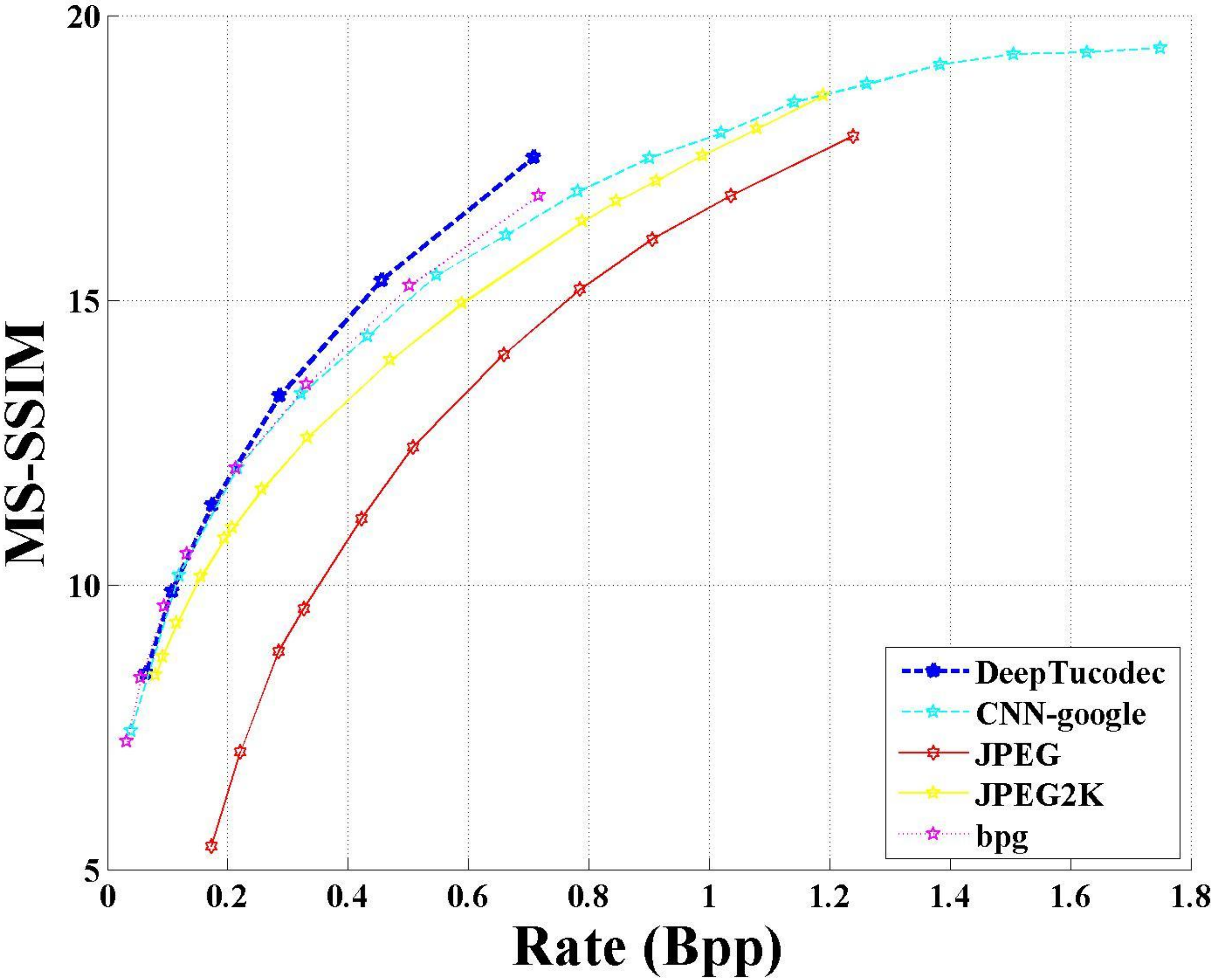
- 我们能做到什么水平？

利用深度学习实现了图片压缩方法**DeepTucodec**

◆ 在KODAK数据集上的测评结果



◆ 在KODAK数据集上的测评结果



◆ 在KODAK数据集上的测评结果-与JPEG比较

后面进入的图为DeepTucodec的结果

◆ 在KODAK数据集上的测评结果-与JPEG2K比较

后面进入的图为DeepTucodec的结果

◆ 在KODAK数据集上的测评结果-与BPG比较

后面进入的图为DeepTucodec的结果

◆ 发展趋势

1改进网络结构和相关的码字估计、熵编码等技术，提高PSNR，进一步改进显示质量

2 为特定的应用场景定制高效的算法

3 设计端到端的视频编码结构

4 将编码与分割、分类、检索等多任务相结合

谢谢！

QA

此页开始 往后所有 PPT 页面

底部低栏 不能遮盖

谢谢合作