

فشرده‌سازی اطلاعات

۰۱-۷۰۲-۱۰-۱۴۰

بخش چهارم



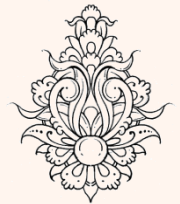
دانشگاه شهید بهشتی
پژوهشکده‌ی فضای مجازی

بهار ۱۳۹۸

احمد محمودی ازناوه

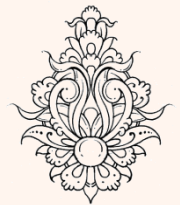
فهرست مطالب

- مقدمه
- فشرده‌سازی بی‌اتلاف
 - فشرده‌سازی بر اساس پیش‌بینی
 - استاندارد قدیمی JPEG
 - فشرده‌سازی CALIC
 - فشرده‌سازی JPEG-LS
 - فشرده‌سازی JBIG
 - کدگذاری Bit-Plane
- فشرده‌سازی با‌تلاف
 - چندی‌سازی بر اساس نمایش بیتی
 - فشرده‌سازی با‌تلاف بر اساس پیش‌بینی و چندی‌سازی



دیباچه

- تاکنون در خصوص روش‌های فشرده‌سازی به صورت کلی بحث شد، هر چند برخی برای کاربردهای خاص مناسب‌تر بوده‌اند، اما محدود به کاربرد خاصی نبودیم.
- در برخی کاربردها، روش‌هایی خاص وابسته به آن کاربرد مورد استفاده قرار می‌گیرند.
- در این بخش در خصوص **فشرده‌سازی** تصویر صحبت خواهد شد.
- یکی از چالش‌های روش‌های آماری در فشرده‌سازی تصویر، بزرگی الفبای مورد استفاده است.



هر چه احتمال وقوع نماد دارای عدم توازن بیشتر باشد، آنتروپی و پیرو آن میزان فشرده‌سازی افزایش می‌یابد.

فشرده‌سازی بر اساس پیش‌بینی

• برای فشرده‌سازی دنباله‌ی زیر،

1	2	5	7	2	-2	0	-5	-3	-1	1	-2	-7	-4	-2	1	3	4
---	---	---	---	---	----	---	----	----	----	---	----	----	----	----	---	---	---

• چنانچه احتمال وقوع نمادها یکسان باشد به چهار بیت برای فشرده‌سازی نیاز است.

• در صورتی که قانون پیش‌بینی «نماد پیشین به اضافه‌ی ۲» باشد، خواهیم داشت:

1	-1	1	0	-7	-4	0	-7	0	0	0	-5	-7	1	0	1	0	-1
---	----	---	---	----	----	---	----	---	---	---	----	----	---	---	---	---	----

– با استفاده از این شیوه‌ی مدل کردن، احتمال وقوع نمادهای چون «0» و «-1» افزایش خواهد یافت.

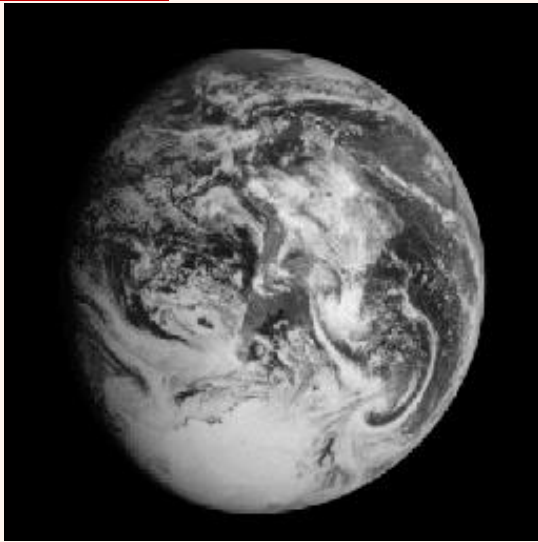
– آنتروپی مقادیر باقیمانده کمتر از آنتروپی مقادیر اصلی است.

- در نتیجه با تعداد بیت کمتری قابل نمایش خواهد بود.
- با توجه به این که فضا براساس نماد قبلی به دست می‌آید، کدگشا به راحتی یا در اختیار داشتن نماد قبلی و میزان فضا می‌تواند نماد اصلی را به دست آورد.



256×256

تصاویر نمونه



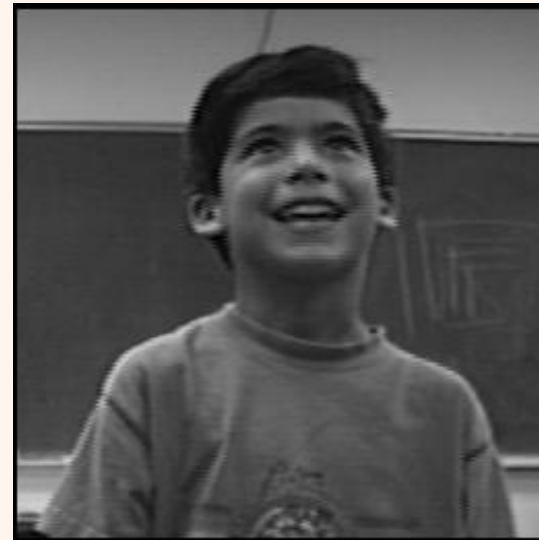
Earth



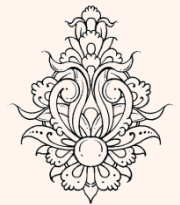
Omaha



Sensin



Sena



استاندارد قدیمی JPEG

The Joint Photographic Experts Group (JPEG)

• در استاندارد قدیمی JPEG هشت شیوهی پیش‌بینی در نظر گرفته شده است:

– در اولین شیوه هیچ پیش‌بینی انجام نمی‌شود و سایر

شیوه‌ها به قرار زیر می‌باشند:

1 $\hat{I}(i, j) = I(i - 1, j)$

2 $\hat{I}(i, j) = I(i, j - 1)$

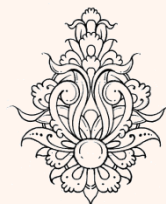
3 $\hat{I}(i, j) = I(i - 1, j - 1)$

4 $\hat{I}(i, j) = I(i, j - 1) + I(i - 1, j) - I(i - 1, j - 1)$

5 $\hat{I}(i, j) = I(i, j - 1) + (I(i - 1, j) - I(i - 1, j - 1)) / 2$

6 $\hat{I}(i, j) = I(i - 1, j) + (I(i, j - 1) - I(i - 1, j - 1)) / 2$

7 $\hat{I}(i, j) = (I(i, j - 1) + I(i - 1, j)) / 2$



تصاویر مختلف ساختارهای متفاوتی دارند، هر نوع پیش‌بینی برای یک ساختار خاص مناسب‌تر است، برای یک تصویر خاص می‌توان همه‌ی شیوه‌ها را بررسی و بهترین را ارسال کرد. فضای پیش‌بینی برای ذخیره‌سازی (ارسال) مورد استفاده قرار می‌گیرد.

مثال

1 $\hat{I}(i, j) = I(i - 1, j)$

2 $\hat{I}(i, j) = I(i, j - 1)$

3 $\hat{I}(i, j) = I(i - 1, j - 1)$

4 $\hat{I}(i, j) = I(i, j - 1) + I(i - 1, j) - I(i - 1, j - 1)$

5 $\hat{I}(i, j) = I(i, j - 1) + (I(i - 1, j) - I(i - 1, j - 1)) / 2$

6 $\hat{I}(i, j) = I(i - 1, j) + (I(i, j - 1) - I(i - 1, j - 1)) / 2$

7 $\hat{I}(i, j) = (I(i, j - 1) + I(i - 1, j)) / 2$

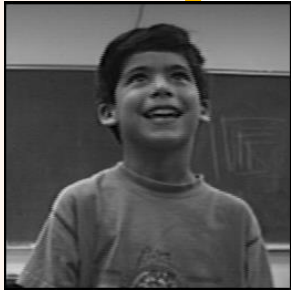
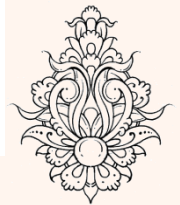


Image	JPEG 0	JPEG 1	JPEG 2	JPEG 3	JPEG 4	JPEG 5	JPEG 6	JPEG 7
Sena	53,431	37,220	31,559	38,261	31,055	29,742	33,063	32,179
Sensin	58,306	41,298	37,126	43,445	32,429	33,463	35,965	36,428
Earth	38,248	32,295	32,137	34,089	33,570	33,057	33,072	32,672
Omaha	56,061	48,818	51,283	53,909	53,771	53,520	52,542	52,189

Image	Best JPEG	GIF	PNG
Sena	31,055	51,085	31,577
Sensin	32,429	60,649	34,488
Earth	32,137	34,276	26,995
Omaha	48,818	61,341	50,185



فشرده‌سازی CALIC

The Context Adaptive Lossless Image Compression (CALIC)

CALIC - A CONTEXT BASED ADAPTIVE LOSSLESS IMAGE CODEC

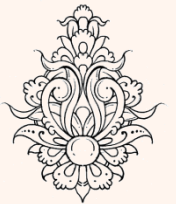
Xiaolin Wu¹

Nasir Memon²

¹Department of Computer Science, The University of Western Ontario, London, Ontario, Canada N6A 5B7F

² Computer Science Department, Northern Illinois University, Dekalb, IL 60115

- در سال ۱۹۹۴ در پاسخ به فراخوان ISO توسط Wu و Memon مطرح شد.
- در مورد مختلف دارد:
 - تصاویر سیاه و سفید (دوسطی)
 - تصاویر طیف فاکستری
- تغییر مورد به صورت خودکار و به صورت محلی صورت می‌پذیرد.
- تصویر را به صورت خط به خط پویش می‌کند.
- در تصویر مقدار یک پیکسل معمولاً به همسایگانش شبیه است.
- شبیه‌ترین همسایه از روی **ساختار** تصویر قابل پیش‌بینی است.



فشرده‌سازی CALIC (ادامه...)

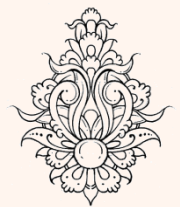
- برای تصاویر چندسطحی دارای چهار بخش است:
 - پیش‌بینی بر اساس گرادیان
 - انتخاب محتوا و چندی‌سازی
 - مدل کردن محتوا و خطای پیش‌بینی
 - کدگذاری

gradient-adjusted prediction (GAP),

context selection and quantization

context modeling of prediction errors,

entropy coding of prediction errors



پیش بینی اولیه

گرادیان در جهت افقی(لبه‌ی عمودی)

$$d_h = |W - WW| + |N - NW| + |N - NE|$$

گرادیان در جهت عمودی(لبه‌ی افقی)

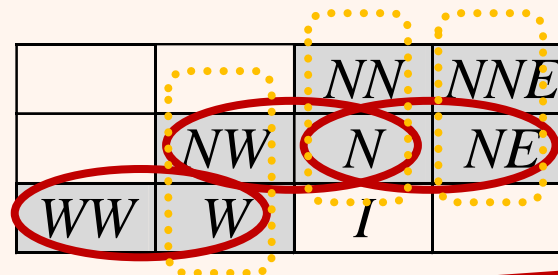
$$d_v = |W - NW| + |N - NN| + |NE - NNE|$$

اختلاف گرادیان عمودی و افقی

$$d = d_v - d_h$$

```

IF ( $d_v - d_h > 80$ ) {sharp horizontal edge}
     $\hat{I}[i, j] = w$ 
ELSE IF ( $d_v - d_h < -80$ ) {sharp vertical edge}
     $\hat{I}[i, j] = n$ 
ELSE {
     $\hat{I}[i, j] = (w + n)/2 + (ne - nw)/4$ ;
    IF ( $d_v - d_h > 32$ ) {horizontal edge}
         $\hat{I}[i, j] = (\hat{I}[i, j] + w)/2$ 
    ELSE IF ( $d_v - d_h > 8$ ) {weak horizontal edge}
         $\hat{I}[i, j] = (3\hat{I}[i, j] + w)/4$ 
    ELSE IF ( $d_v - d_h < -32$ ) {vertical edge}
         $\hat{I}[i, j] = (\hat{I}[i, j] + n)/2$ 
    ELSE IF ( $d_v - d_h < -8$ ) {weak vertical edge}
         $\hat{I}[i, j] = (3\hat{I}[i, j] + n)/4$ 
    }
    
```



ضرائب پیش بینی و حد آستانه‌ها به صورت تجربی انتخاب شده‌اند و یکی از اصول مهم در انتخاب سادگی محاسبات بوده است، از این جهت بیشتر ضریب‌ها توانی از دو در نظر گرفته شده‌اند.

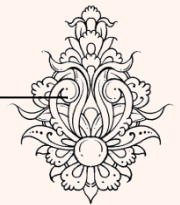
برای دسته‌های فاص از تصاویر می‌توان به صورت جداگانه ضرایب بهینه را به دست آورد.

مثال ۱

\hat{I}

		<i>NN</i>	<i>NNE</i>
	<i>NW</i>	<i>N</i>	<i>NE</i>
<i>WW</i>	<i>W</i>	<i>I</i>	

<i>d</i>	<table border="1"><tr><td></td><td></td><td>200</td><td>200</td></tr><tr><td></td><td>200</td><td>200</td><td>200</td></tr><tr><td>100</td><td>100</td><td><i>I</i></td><td></td></tr></table>			200	200		200	200	200	100	100	<i>I</i>		Sharp horizontal edge	<i>W</i>
		200	200												
	200	200	200												
100	100	<i>I</i>													
80	<table border="1"><tr><td></td><td></td><td>160</td><td>160</td></tr><tr><td></td><td>160</td><td>160</td><td>160</td></tr><tr><td>100</td><td>100</td><td><i>I</i></td><td></td></tr></table>			160	160		160	160	160	100	100	<i>I</i>		horizontal edge	$(t+W)/2$
		160	160												
	160	160	160												
100	100	<i>I</i>													
32	<table border="1"><tr><td></td><td></td><td>130</td><td>130</td></tr><tr><td></td><td>130</td><td>130</td><td>130</td></tr><tr><td>100</td><td>100</td><td><i>I</i></td><td></td></tr></table>			130	130		130	130	130	100	100	<i>I</i>		weak horizontal edge	$(3t+W)/4$
		130	130												
	130	130	130												
100	100	<i>I</i>													
8	<table border="1"><tr><td></td><td></td><td>100</td><td>100</td></tr><tr><td></td><td>100</td><td>100</td><td>100</td></tr><tr><td>100</td><td>100</td><td><i>I</i></td><td></td></tr></table>			100	100		100	100	100	100	100	<i>I</i>		homogeneous	<i>t</i>
		100	100												
	100	100	100												
100	100	<i>I</i>													
-8	<table border="1"><tr><td></td><td></td><td>100</td><td>100</td></tr><tr><td></td><td>130</td><td>100</td><td>100</td></tr><tr><td>130</td><td>130</td><td><i>I</i></td><td></td></tr></table>			100	100		130	100	100	130	130	<i>I</i>		weak vertical edge	$(3t+N)/4$
		100	100												
	130	100	100												
130	130	<i>I</i>													
-32	<table border="1"><tr><td></td><td></td><td>100</td><td>100</td></tr><tr><td></td><td>160</td><td>100</td><td>100</td></tr><tr><td>160</td><td>160</td><td><i>I</i></td><td></td></tr></table>			100	100		160	100	100	160	160	<i>I</i>		vertical edge	$(t+N)/2$
		100	100												
	160	100	100												
160	160	<i>I</i>													
-80	<table border="1"><tr><td></td><td></td><td>100</td><td>100</td></tr><tr><td></td><td>200</td><td>100</td><td>100</td></tr><tr><td>200</td><td>200</td><td><i>I</i></td><td></td></tr></table>			100	100		200	100	100	200	200	<i>I</i>		Sharp vertical edge	<i>N</i>
		100	100												
	200	100	100												
200	200	<i>I</i>													



$$t = \frac{W + N}{2} + \frac{NE - NW}{4}$$

Guu-In Chen

فشرده سازی



مثال ۲

		<i>NN</i>	<i>NNE</i>
	<i>NW</i>	<i>N</i>	<i>NE</i>
<i>WW</i>	<i>W</i>	<i>I</i>	

		200	100
	200	100	100
200	100	<i>I</i>	

$$d_h = |100 - 200| + |100 - 200| + |100 - 100| = 200$$

$$d_v = |100 - 200| + |100 - 200| + |100 - 100| = 200$$

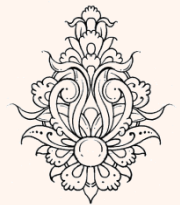
$$d = d_v - d_h = 200 - 200 = 0$$

$$\hat{I} = t = \frac{W + N}{2} + \frac{NE - NW}{4} = \frac{100 + 100}{2} + \frac{100 - 200}{4} = 100 - 25 = 75$$

$$e = I - \hat{I}$$

خطای پیش‌بینی

If $I=100$ then $e=100-75=25$



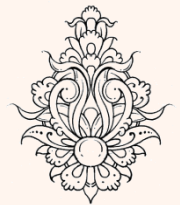
انتخاب محتوا و چندی سازی

- مراحل انجام شده، «افزونی آماری» موجود را کاملاً برطرف نمی‌کند.
- آزمایش‌ها نشان می‌دهد واریانس خطای به دست آمده با میزان همواری تصاویر رابطه‌ی مستقیم دارد.
- برای یک تخمین با محاسبات ساده رابطه‌ی زیر مورد استفاده قرار گرفته است:

$$\Delta = d_h + d_v + 2|e_w|$$

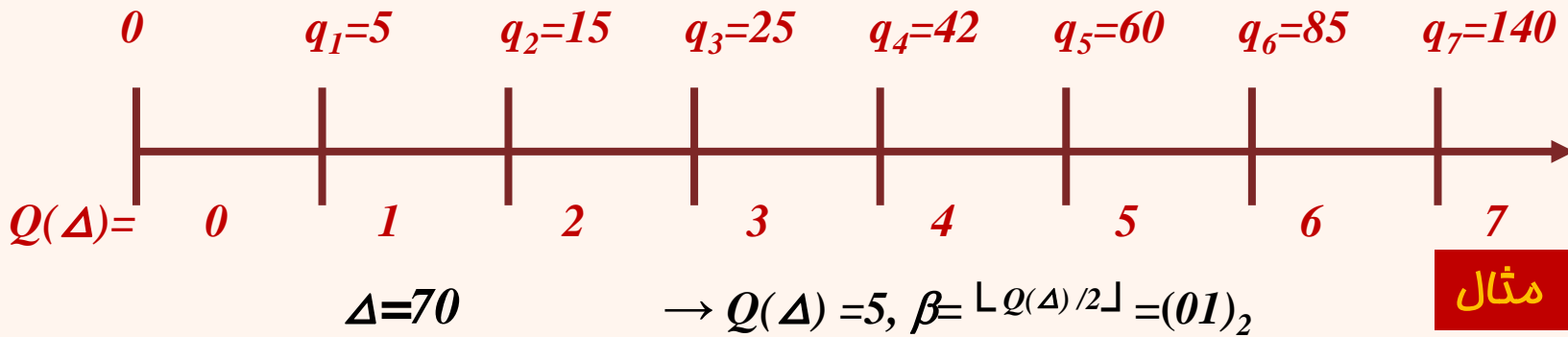
$$|e_w| = |W - \hat{W}|$$

خطای پیش‌بینی قبلی: خطا معمولاً به صورت متوالی رخ می‌دهد.



انتخاب محتوا و چندی سازی

- کدگذاری با استفاده از احتمال شرطی $p(e|\Delta)$ کارا تر خواهد بود.
- برای صرفه جویی در مکان و زمان، Δ به L سطح چندی می شود ($L=8$).



- این ضرایب به گونه ای انتخاب می شوند که رابطه ی زیر کمینه شود (اعداد فوق با آزمایش به دست آمده اند):

$$-\sum_e p(e) \log p(e|q_d \leq \Delta < q_{d+1})$$



مدل کردن محتوا و خطای پیش‌بینی

- در نظر گرفتن بافت تصویر می‌تواند منجر به ارائه‌ی مدل دقیق‌تری شود.

$$C = \{N, W, NW, NE, NN, WW, 2N-NN, 2W-WW\}$$

$$= \{x_0, x_1, x_2, x_3, x_4, x_5, x_6, x_7\}$$

$$b_k = \begin{cases} 0 & \text{if } x_k \geq \hat{I} \\ 1 & \text{if } x_k < \hat{I} \end{cases}$$

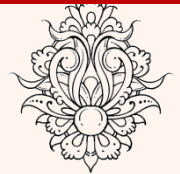
$$\alpha = b_7 b_6 \dots b_0$$

		200	100
	200	100	100
200	100	I	

		NN	NE
	NW	N	NE
WW	W	I	

مثال

$$\hat{I} = 75$$



$$C = \{100, 100, 200, 100, 200, 200, 0, 0\}$$

$$b_{0 \sim 7} = 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 1$$

$$\alpha = 1100000_2$$



انتخاب محتوی

• بدین ترتیب α و β محتوا را مشخص می‌کنند.

α and $\beta \Rightarrow C(\alpha, \beta)$

$\alpha=11000000$

$\beta=10$

$C(\alpha, \beta)=11000000\underline{10}$

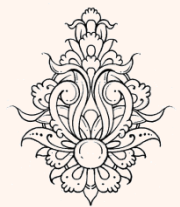
مثال

با توجه به $C(\alpha, \beta)$ پیکسل‌های به دست آمده را تقسیم‌بندی می‌کنیم.

داده‌های مربوط به هر بخش (محتوا) به صورت جداگانه کد می‌شوند.

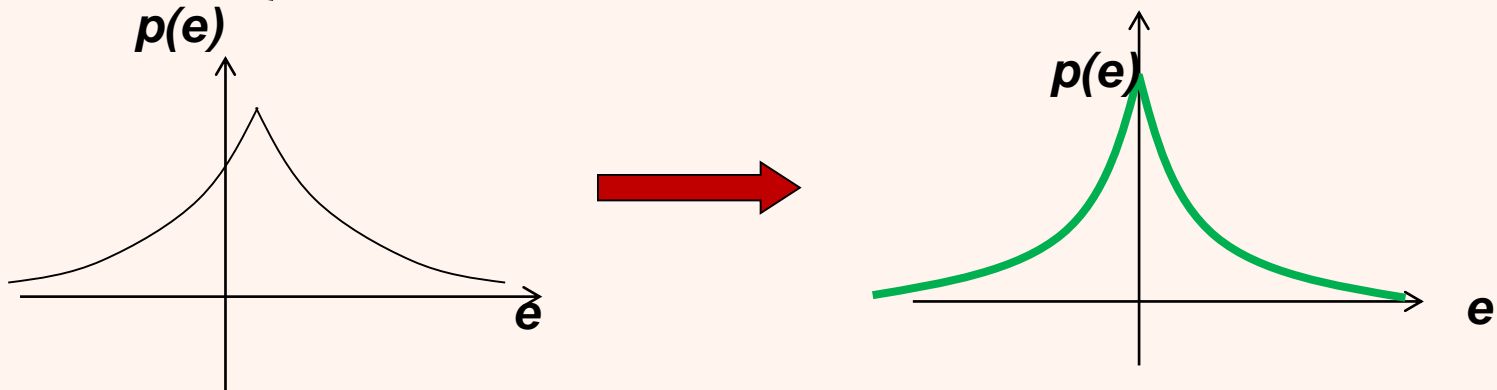
تعداد کل حالات برابر است با

$$\text{cases in } C(\alpha, \beta) = 144 * 4 = 576$$



انتخاب محتوا و چندی سازی

- در اکثر مواقع این پیش بینی داری سوگیری است.
– برای کدگذاری بهتر، باید این مقدار اصلاح شود.



- برای هر دسته $C(\alpha, \beta)$ مقدار زیر حساب می شود:

$$\bar{e}(\alpha, \beta) = \frac{\sum e(\alpha, \beta)}{N(\alpha, \beta)}$$

میانگین فضای دسته

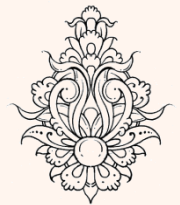
- بر اساس محتوا این اصلاح انجام خواهد شد:

$$\tilde{I} = \hat{I} + \bar{e}(\alpha, \beta)$$

پیش بینی اصلاح شده

$$\varepsilon = I - \tilde{I}$$

فضای اصلاح شده



مثال

$$I = 10, 11, 13, 15$$

$$\hat{I} = 8, 10, 13, 16, 14$$

$$\bar{e} = \frac{2+1+0-1+4}{5} = 1.2 = 1$$

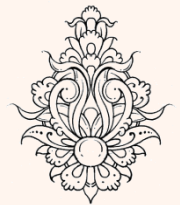
$$e = 2, 1, 0, -1, 4$$

$$\tilde{I} = 9, 11, 14, 17, 15$$

$$\varepsilon = I - \tilde{I} = 1, 0, -1, -2, 3$$

مقایسه‌ی روش‌های فشرده‌سازی بی‌اتلاف تصویر

image	Old JPEG	New JPEG	CALIC
Sena	31,055	27,399	26,433
Sensin	32,429	30,344	29,213
Earth	32,137	26,088	25,280
Omaha	48,818	50,765	48,249



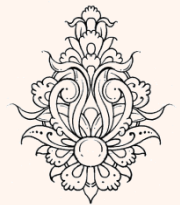
JPG-LS

• در پی‌فراخوان ISO، CALIC در شش مجموعه از بین هفت مجموعه تصویر معیار، رتبه نخست را کسب کرد. با توجه به ویژگی‌های این روش، محققانی از HP شیوه‌ای با حجم محاسبات کمتر و کارایی تقریباً قابل مقایسه به نام **LOCO-I** مطرح کردند، که نهایتاً این شیوه استاندارد شد.

(Low Complexity Lossless COmpression for Images)

NW	N
W	I

$$\hat{I} = \begin{cases} \min(W, N) & \text{if } NW \geq \max(W, N) \\ \max(W, N) & \text{if } NW \leq \min(W, N) \\ W + N - NW & \text{otherwise} \end{cases}$$



NW	N	NE
W	I	

$$\hat{I} = \begin{cases} \min(W, N) & \text{if } NW \geq \max(W, N) \\ \max(W, N) & \text{if } NW \leq \min(W, N) \\ W + N - NW & \text{otherwise} \end{cases}$$

مثال ۱

Median Edge Detector

MED

	0	0
0	0	0
200	I	

horizontal edge

W

200	200	200
200	200	200
0	I	

	100	100
100	100	100
100	I	

homogeneous

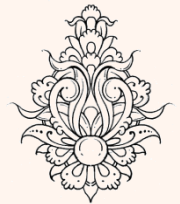
W(or N)

	0	0
200	0	0
200	I	

vertical edge

N

0	200	200
0	200	200
0	I	



به دست آوردن مدل

NW	N	NE
W	I	

$$D_1 = NE - N$$

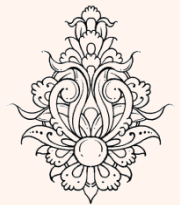
$$D_2 = N - NW$$

$$D_3 = NW - W$$

64	145	145
100	145	

145	145	145
145	145	

$$(D_1, D_2, D_3) = (0, 81, -36) \quad (D_1, D_2, D_3) = (0, 0, 0)$$



چندی سازی

$D_i (i = 1 \text{ to } 3)$	$Q_i (i = 1 \text{ to } 3)$
{0}	0
$\pm\{1,2\}$	± 1
$\pm\{3,4,5,6\}$	± 2
$\pm\{7,8,\dots,20\}$	± 3
$\pm\{\geq 21\}$	± 4

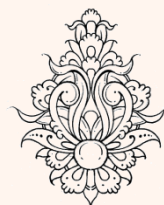
$$(D_1, D_2, D_3) \Rightarrow (Q_1, Q_2, Q_3)$$

$$(D_1, D_2, D_3) = (0, 81, -36)$$

$$\Rightarrow (Q_1, Q_2, Q_3) = (0, 4, -4)$$

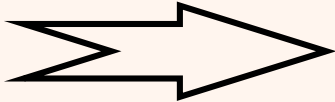
$$(D_1, D_2, D_3) = (0, 0, 0)$$

$$\Rightarrow (Q_1, Q_2, Q_3) = (0, 0, 0)$$



- در این حالت $9 \times 9 \times 9 = 729$ حالت مختلف خواهیم داشت.
- چنانچه در (Q_1, Q_2, Q_3) اولین عنصر ناصفر منفی باشد، این بردار را با $(-Q_1, -Q_2, -Q_3)$ تغییر می‌دهیم.
- با این کار تعداد حالات به ۳۶۵ حالت تقلیل می‌یابد.

Q_1	Q_2	Q_3
0	0	-2
-4	3	-1
0	0	0
0	1	-3
2	2	2



Q_1	Q_2	Q_3
0	0	2
4	-3	1
0	0	0
0	1	-3
2	2	2



شیوه‌های مختلف

• این استاندارد دو شیوه دارد:

– Regular mode

– Run mode

– چنانچه $Q_1=Q_2=Q_3=0$ Run mode انتخاب می‌شود.

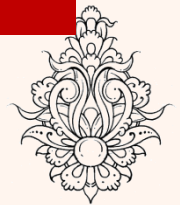
مثال

$$(Q_1, Q_2, Q_3) = (0, 4, -4)$$

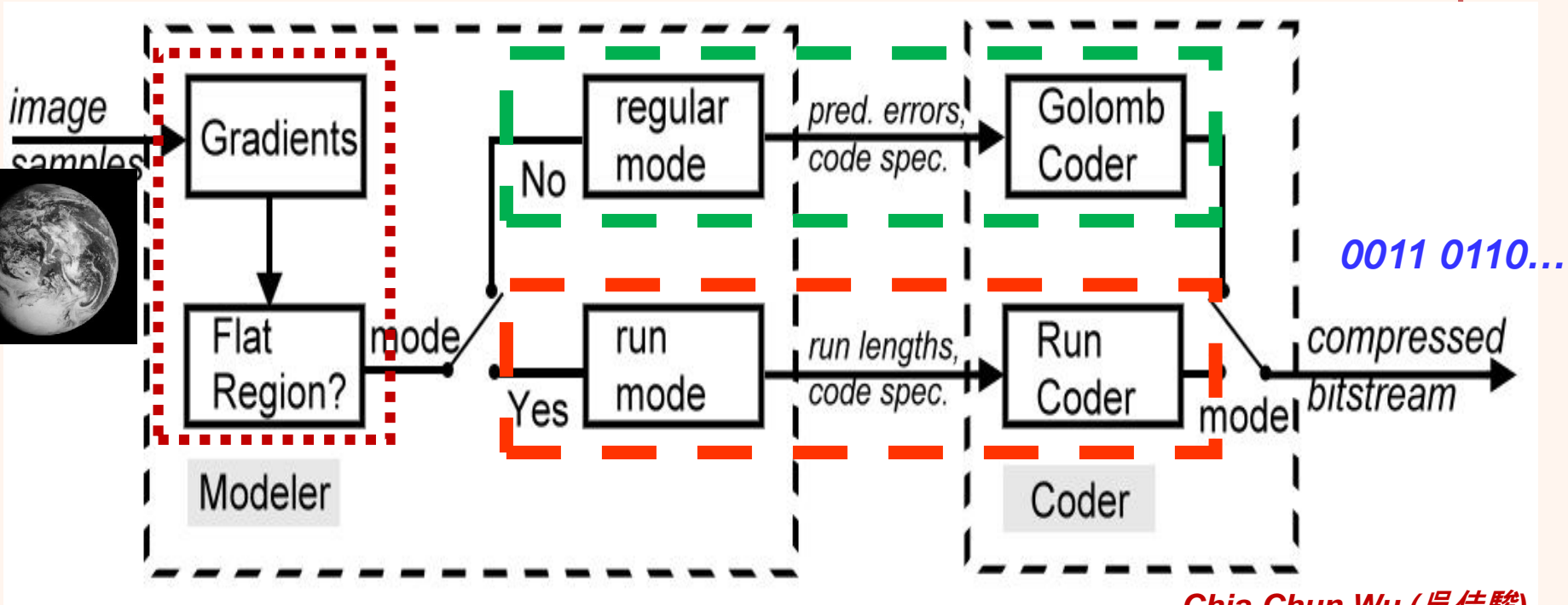
⇒ Regular mode

$$(Q_1, Q_2, Q_3) = (0, 0, 0)$$

⇒ Run mode



دیباگراہ کلی



Chia-Chun Wu (吳佳駿)

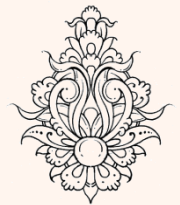


- در این شیوه، قابلیت کد کردن تصویر با وضوح متغیر وجود دارد.
- در این شیوه تصویر با یک ساختار هرمی مدل می‌شود.

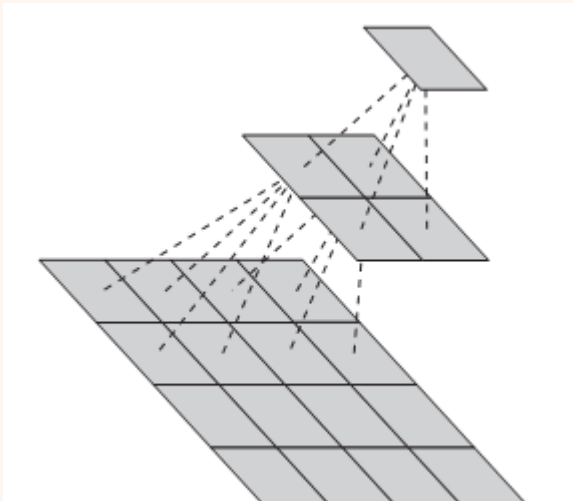
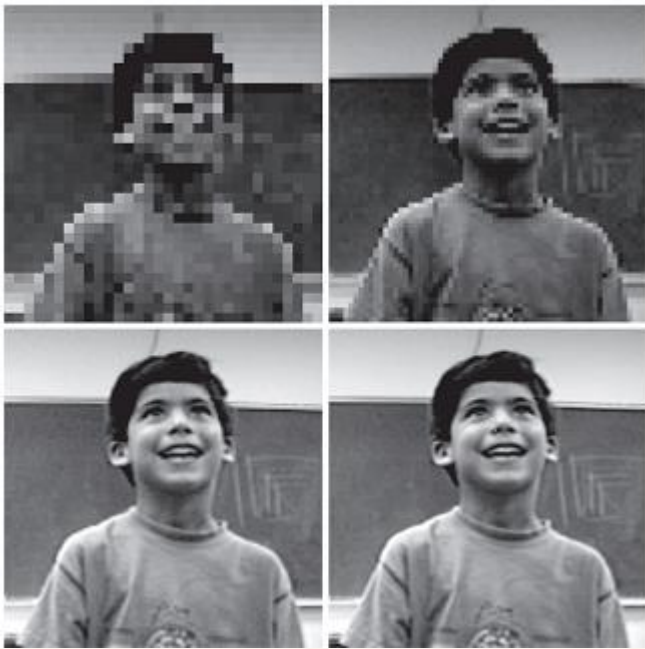
Δ	.	X	.	Δ	.	X	.	Δ
.	*	.	*	.	*	.	*	.
X	.	o	.	X	.	o	.	X
.	*	.	*	.	*	.	*	.
Δ	.	X	.	Δ	.	X	.	Δ
.	*	.	*	.	*	.	*	.
X	.	o	.	X	.	o	.	X
.	*	.	*	.	*	.	*	.
Δ	.	X	.	Δ	.	X	.	Δ

HINT (Hierarchical INTerpolation)

- ابتدا پیکسل‌های Δ بر اساس پیش‌بینی کد می‌شود.
- سپس o با درون‌یابی
- سپس X با کمک دو نوع قبلی
- و در نهایت • و * با کمک همسایگان



progressive image transmission

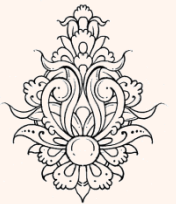


progressive v.s. raster scan



The pyramid structure for progressive transmission

- توسط ITU-T روش‌های متعددی برای فشرده‌سازی دستگاه دورنگار معرفی شده است.
- با گسترش شبکه‌ی اینترنت توجهات بیشتر محطوف به استانداردهای پروتکل‌ها شد.
- در استاندارد اولیه‌ی آنالوگ (بدون امکان فشرده‌سازی) انتقال یک صفحه‌ی A4 مدود شش دقیقه زمان نیاز داشت!
- در فشرده‌سازی این‌گونه اسناد، بیشتر از RLE استفاده می‌شده است.



Joint Bi-Level Image Experts Group

• در این استاندارد امکان کدگذاری به هر دو شیوهی
ترتیبی و پیش‌رونده وجود دارد.

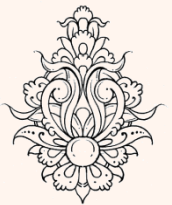
– در شیوهی ترتیبی تصویر از چپ به راست و بالا به
پایین پویش می‌شود.

– در شیوهی پیش‌رونده تصاویر با وضوح متفاوت
کدگذاری می‌شوند.

– امکان بهره‌مندی از هر دو شیوه به صورت توأم هم
وجود دارد.

progressive compatible sequential mode

• در این شیوه تصویر به یک‌سری نوار تقسیم می‌شود. هر نوار
به صورت پیش‌رونده و نوارها به صورت ترتیبی کد می‌شوند.



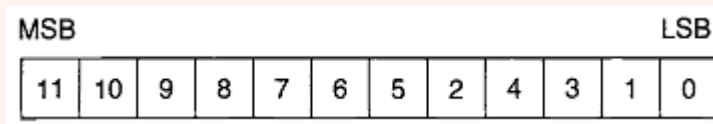
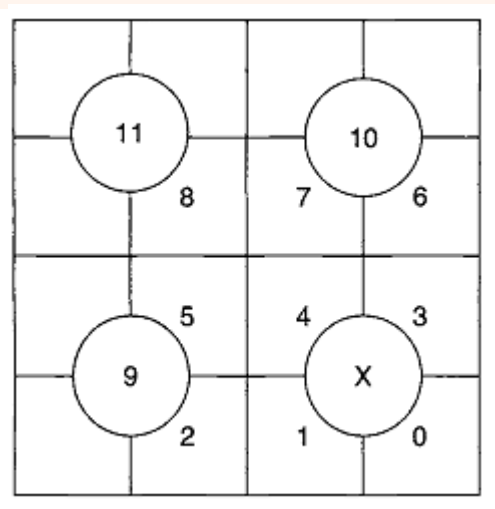
تصاویر دوسطحی

- مراحل مختلف این الگوریتم به صورت زیر است:
 - کاهش وضوح (در شیوهی پیش‌رونده)
 - کاهش افزونگی
 - کدگذاری محاسباتی

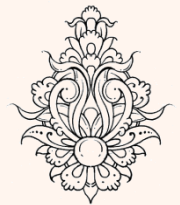


Resolution Reduction

- در این استاندارد یک لایه با وضوح کمتر، تصویری است که در آن ابعاد تصویر یک چهارم شده است.
- برای تعداد لایه‌ها محدودیتی در نظر گرفته نشده است.
- با توجه به همسایه‌ها و یک جدول مراجعه (۱×۹۶×۱۴۰) مقدار پیکسل‌ها در وضوح کمتر به دست می‌آیند.



آدرس مقدار مناسب برای X
با توجه به آدرس به دست
آمده از جدول استخراج
می‌شود.



Redundancy Removal

typical prediction

deterministic prediction

• دو نوع پیش‌بینی وجود دارد:

– در تصاویر دوسطحی بسیاری از نواحی دارای سطح یکسان هستند.

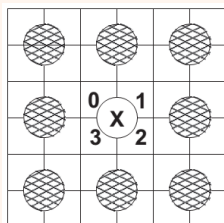
• به ویژه نواحی زیادی سفید هستند.

– برای این نواحی در دو تصویر با وضوح پایین و بالا، پیکسل‌ها همسان هستند.

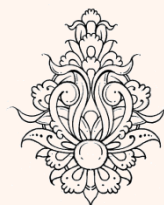
– در این حالت با در اختیار داشتن تصاویر با وضوح پایین در این نواحی، لزومی ندارد تصاویر با وضوح بالا هم جداگانه کد شوند.

– در این حالت در صورتی که یک پیکسل در وضوح پایین در ناحیه‌ای هموار قرار داشته باشد (تمام هشت همسایه‌اش یکسان باشد)، و همچنین با هر چهار همسایه با وضوح بالای خودش هم یکسان باشد، این پیکسل typical نامیده می‌شود.

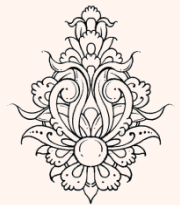
– در صورتی که یک سطر شامل پیکسل‌های این‌چنینی باشد (با یک **پرچه** مشخص می‌شود)، از کد کردن آن صرف‌نظر می‌شود.

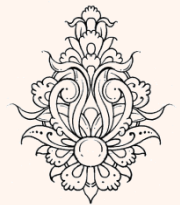
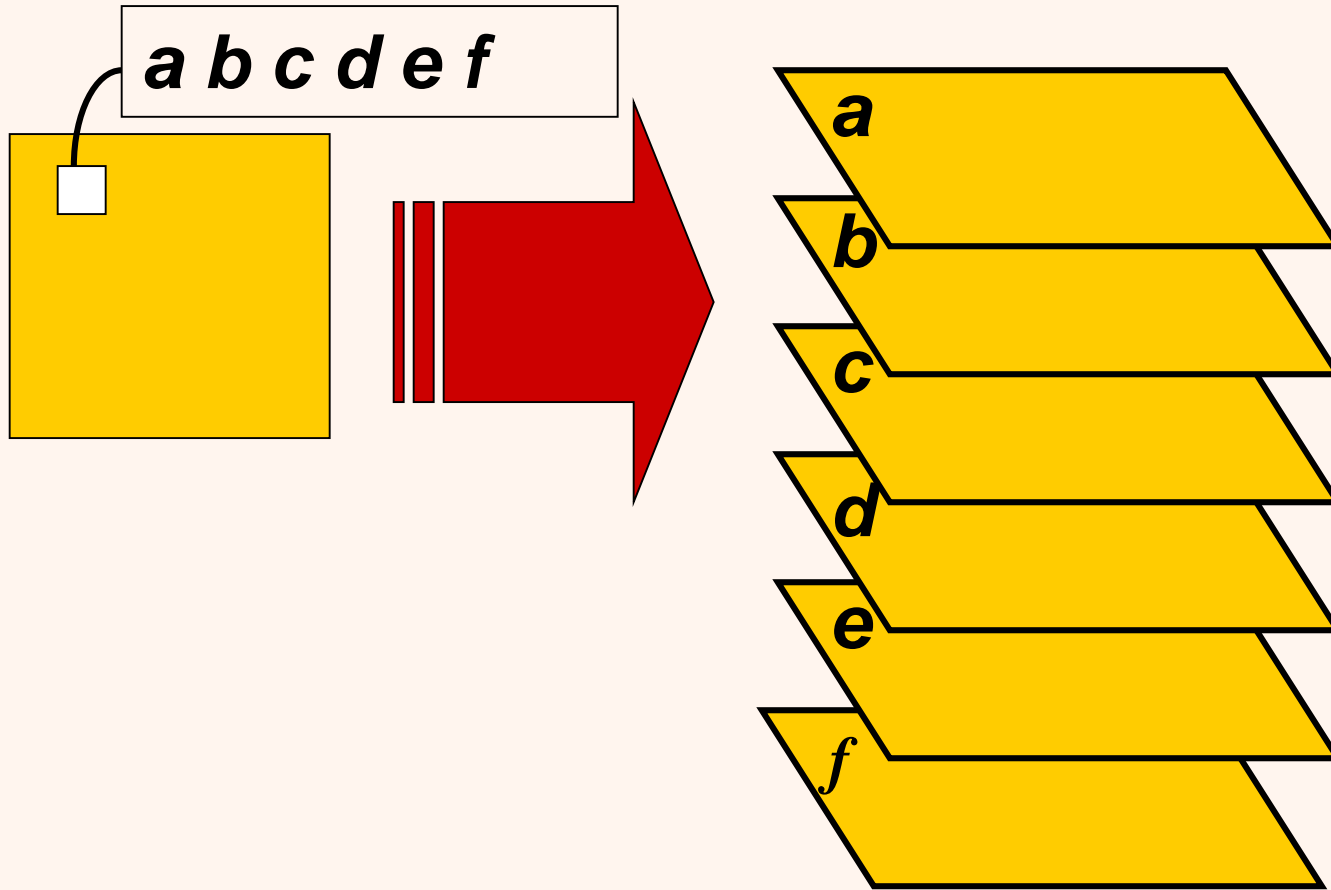


LNTP (line not typical)



- با توجه به جدولی که برای کاهش وضوح مورد استفاده قرار گرفته است، می‌توان در برخی موارد مقدار یک پیکسل را از روی نسخه‌ی با وضوح پایین پیش‌بینی کرد، با این کار می‌توان تا حدود ۷٪ فشردگی به دست آورد.





نمایش بیتی

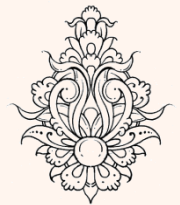
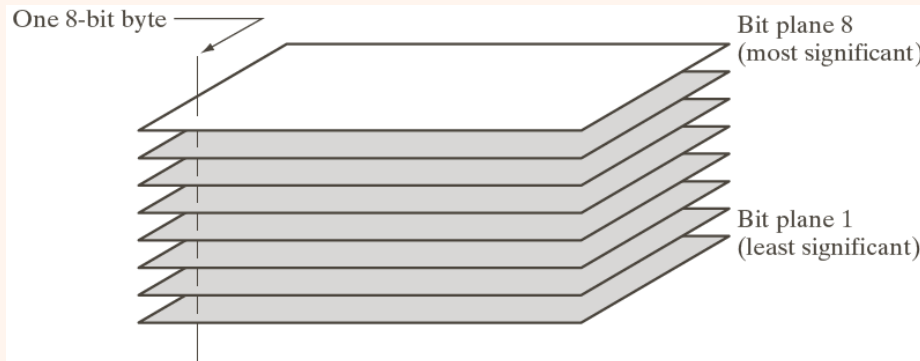
Bit Extraction

- به نمایش مقادیر روشنایی به صورت لایه‌هایی از تصاویر دودویی گفته می‌شود.
- با فرض داشتن $L=256$ داریم:

بیت پر ارزش

$$r = b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0 = \sum_{i=0}^7 b_i 2^i$$

- متناظر با هر بیت یک تصویر ساخته می‌شود.



- در این حالت می‌توان تصاویر دوسطحی را به صورت مجزا با الگوریتم‌های مرتب‌فشارده کرد.
- در این حالت می‌توان از ارتباط بین تصاویر دودویی هم استفاده کرد.



تصویر اصلی

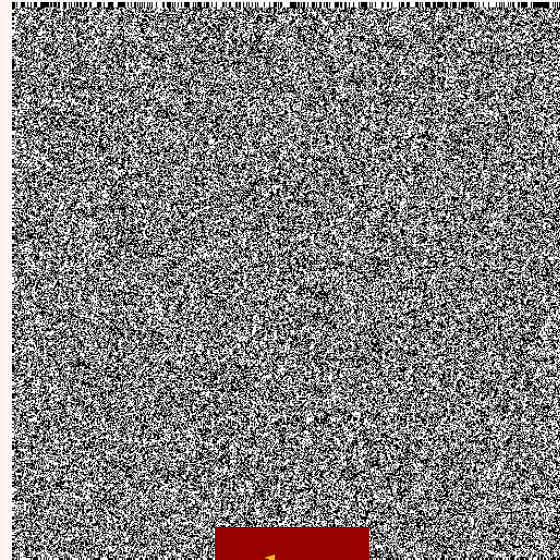
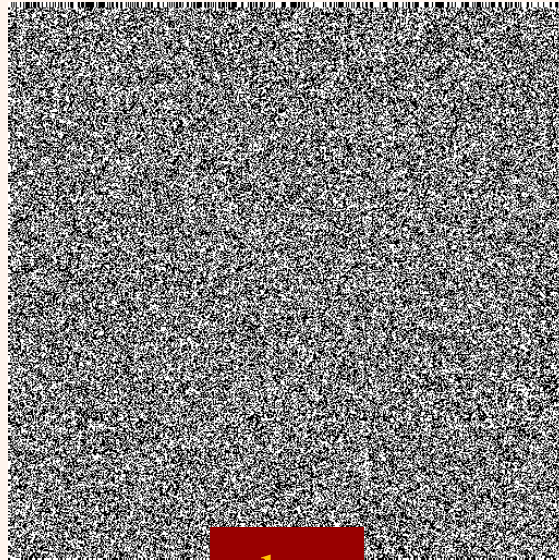
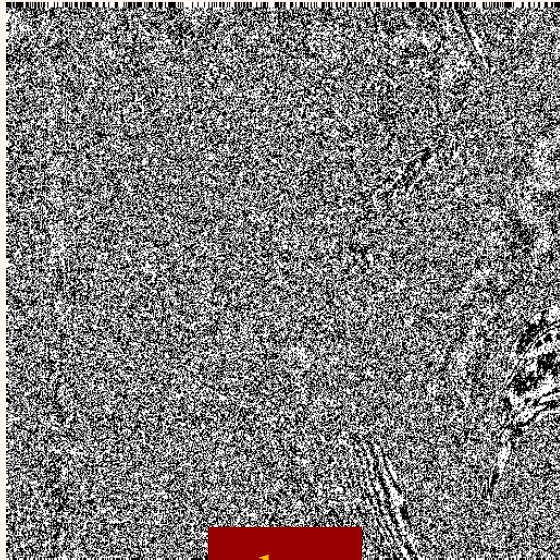
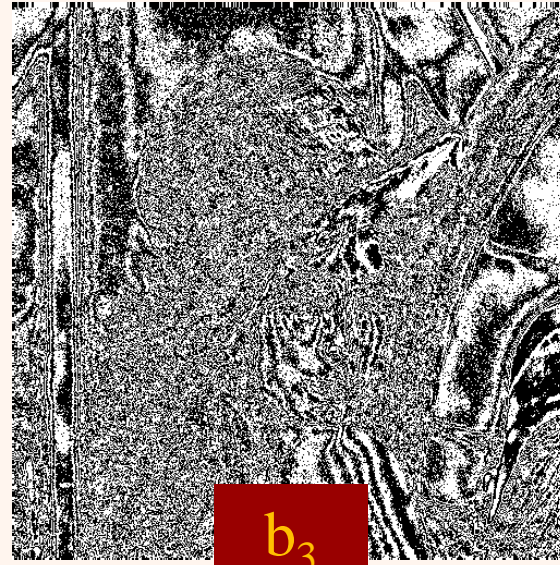
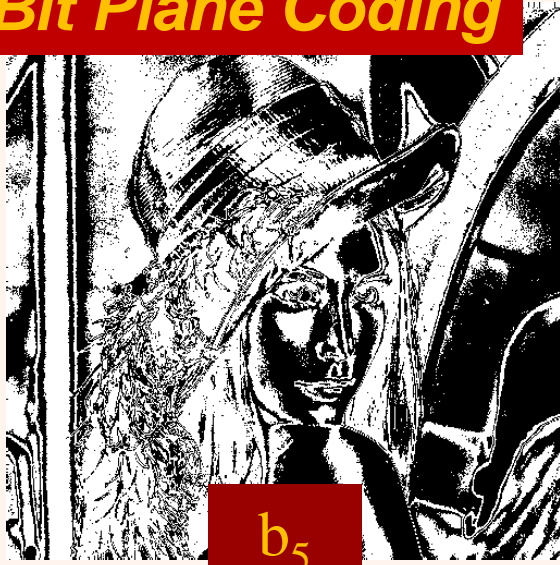


b_7



b_6

Bit Plane Coding



شاهد
شهرت
بهیشتی

تصاویر به دست آمده بعضا همبستگی پایینی دارند!



فشرده‌سازی
باتلاف با
چندی‌سازی

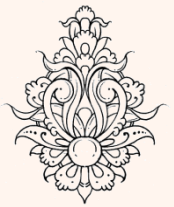


$b_7b_6b_5b_4b_3000$

$b_7b_6b_500000$



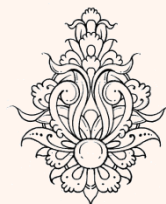
$b_7b_6b_5b_40000$



تراشگاه
سپیدی
بهشتی

نمایش دودویی و Gray

- بیت‌های کم‌ارزش همانند **نویز سفید** خواهند بود و می‌توان از آن‌ها صرف‌نظر کرد.
 - جهت نمایش مدل bit-plane علاوه بر **کد دودویی** از **کدگذاری Gray** استفاده می‌شود.
 - استفاده از **کد gray** به این دلیل است که در نمایش دودویی کوچک‌ترین تغییر در مقدار روشنایی باعث می‌شود پیکسل‌های متناظر شباهتی به هم نداشته باشند.
- به طور مثال دو پیکسل مجاور با اندازه‌ی ۱۲۷ و ۱۲۸ با این‌که یک مقدار تفاوت دارند در نمایش دودویی در تمام ۸ بیت اختلاف خواهند داشت.



حل مشکل با کد Gray

Gray code

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

Binary code

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

$$\{b_{m-1}, b_{m-2}, \dots, b_1, b_0\} \longrightarrow \{g_{m-1}, g_{m-2}, \dots, g_1, g_0\}$$

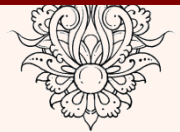
$$g_{m-1} = b_{m-1}$$

$$g_i = b_i \oplus b_{i+1} \quad 0 \leq i \leq m-2$$

$$1110 \longrightarrow 1001$$

$$g_{m-1} = b_{m-1}$$

$$g_i = b_i \oplus b_{i+1} \quad 0 \leq i \leq m-2$$



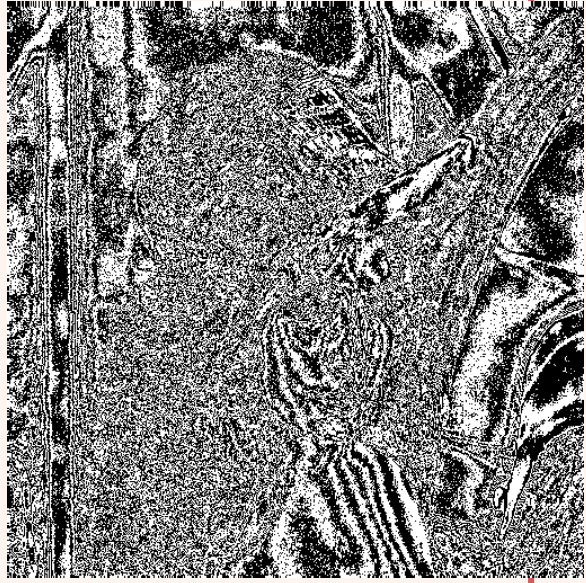
Gray-coded



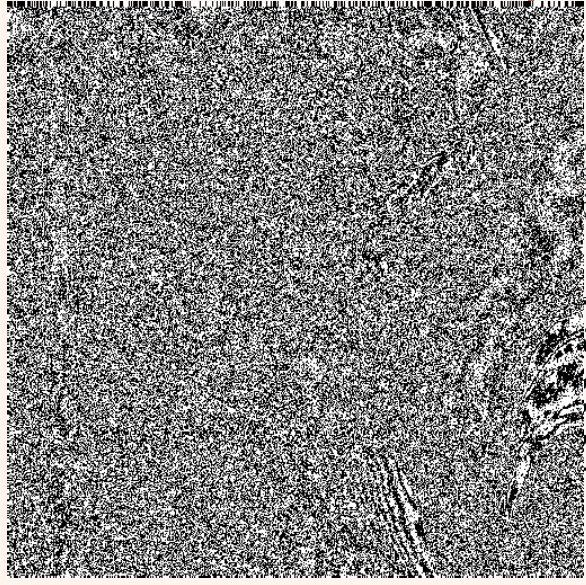
binary



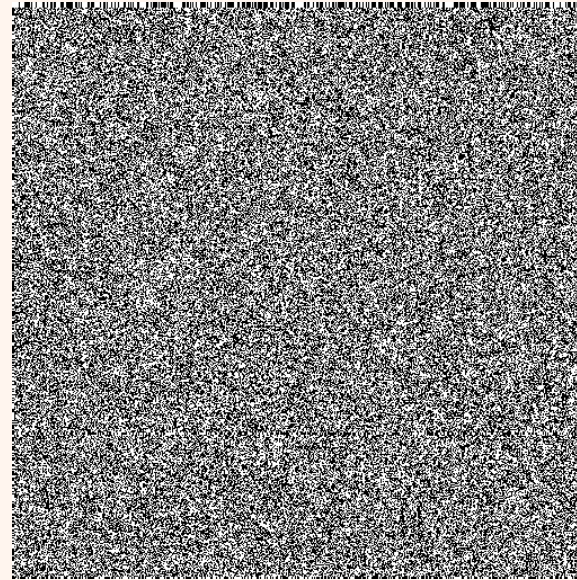
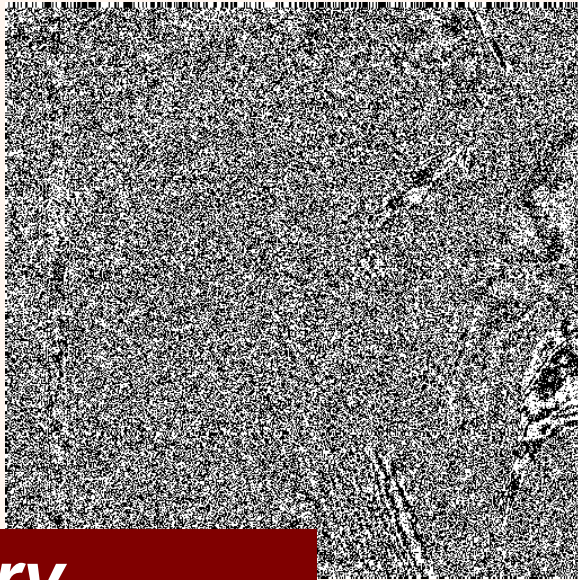
Gray-coded



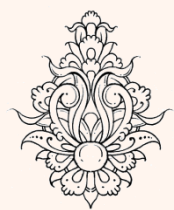
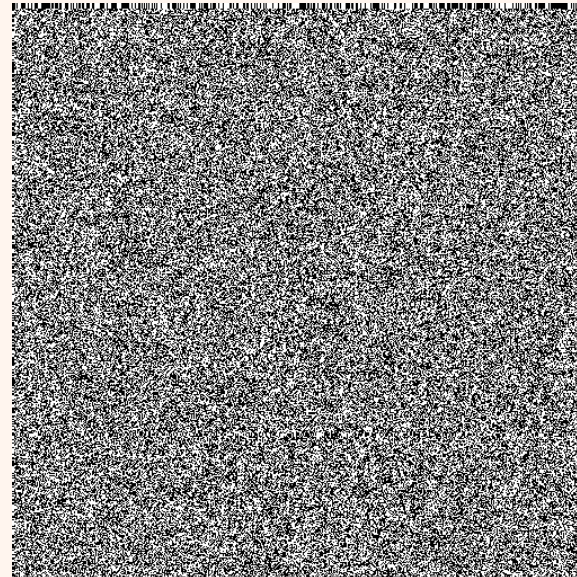
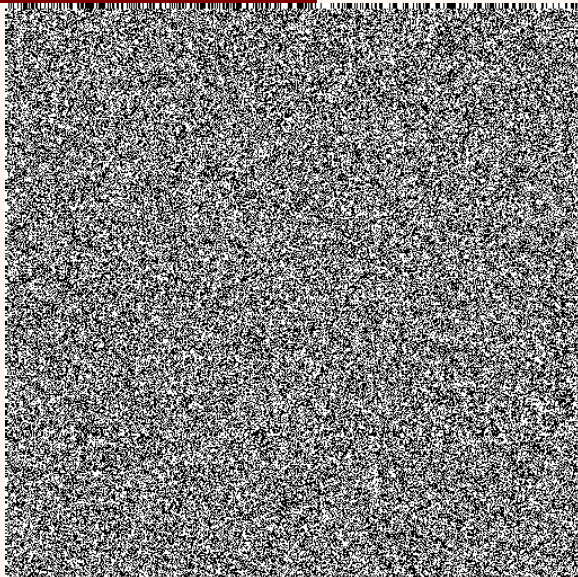
binary



Gray-coded



binary



binary

Gray-coded



All bits



$a_{7,i}$



a_6



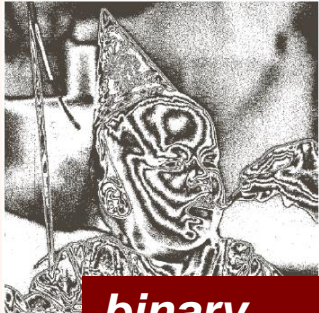
g_6



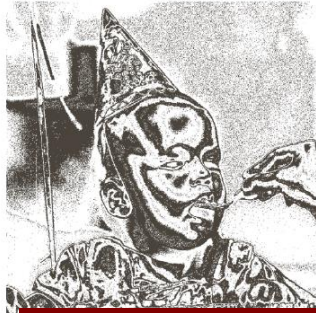
a_5



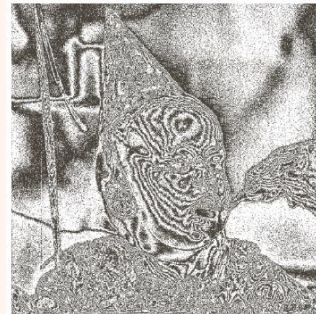
g_5



binary



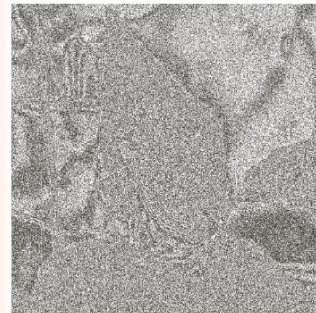
Gray-coded



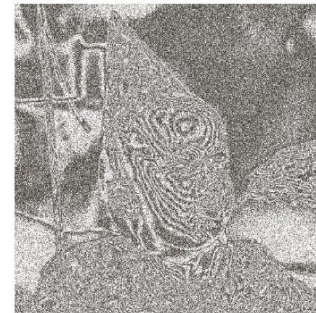
a_3



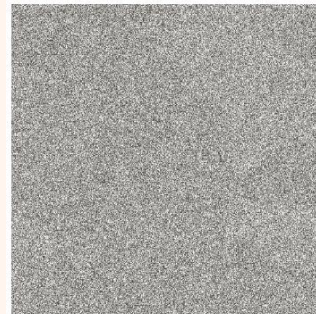
g_3



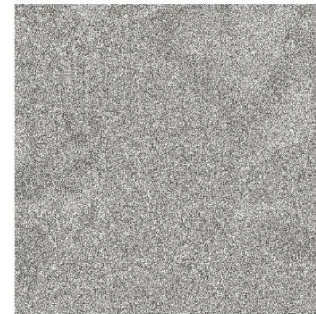
a_2



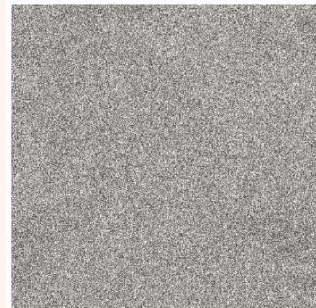
g_2



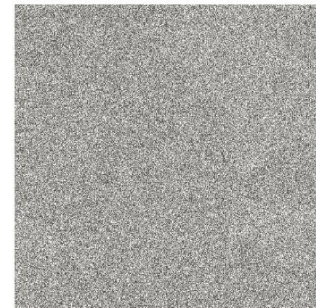
a_1



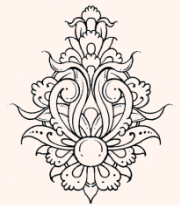
g_1



a_0



g_0



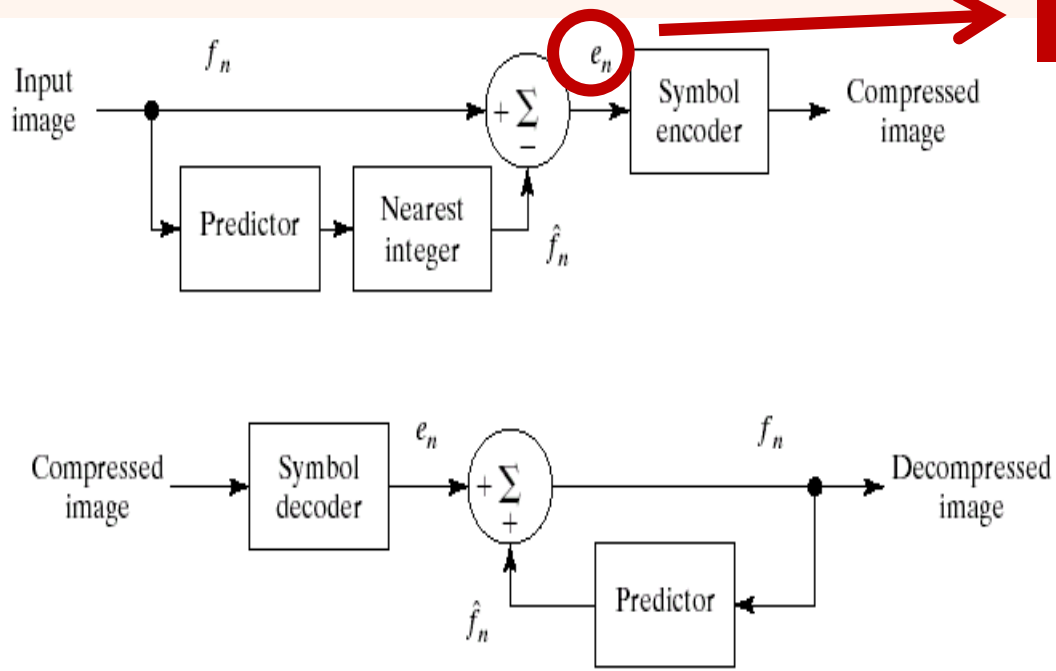
lossless predictive coding

- یکی از راه‌های کاهش آنتروپی و افزایش میزان فشرده‌سازی، بهره‌گیری از افزونگی آماری موجود در تصویر است.

– کد کردن خطای پیش‌بینی پیش از این مطرح شد.

a
b

FIGURE 8.19 A lossless predictive coding model: (a) encoder; (b) decoder.



Q(e)

Lossy!



Lossless predictive coding

$$\begin{array}{rcccccc} n = & 0 & 1 & 2 & 3 & 4 & 5 \\ f(n) = & 100 & 102 & 120 & 120 & 120 & 118 \\ \hat{f}(n) = & - & 100 & 102 & 120 & 120 & 120 \\ e(n) = & - & 2 & 18 & 0 & 0 & -2 \\ \tilde{f}(n) = & 100 & 102 & 120 & 120 & 120 & 118 \end{array}$$

ارسال می‌شود

بدون اتلاف!

$$e_Q(n) = \begin{cases} 4 & 10 < e \\ 3 & 5 < e \leq 10 \\ 2 & 2 < e \leq 5 \\ 1 & 0 < e \leq 2 \\ \vdots & \vdots \end{cases}$$

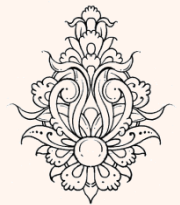
Plus Quantization \rightarrow Lossy coding

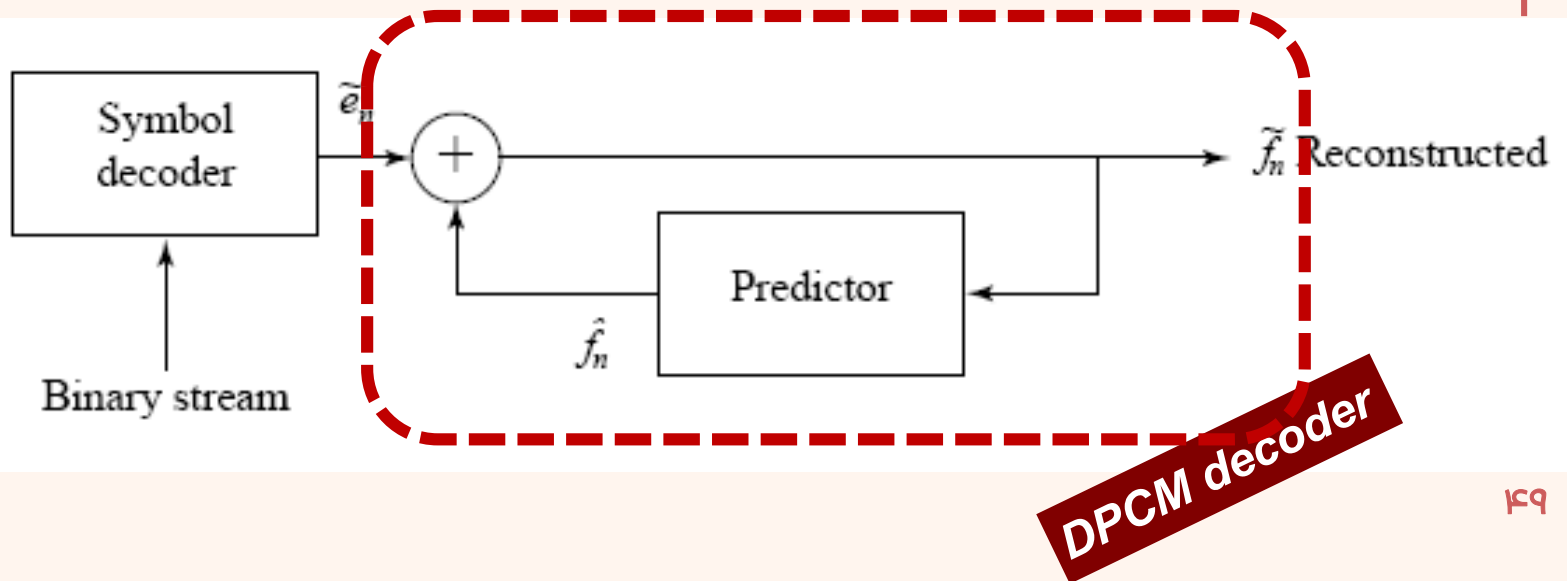
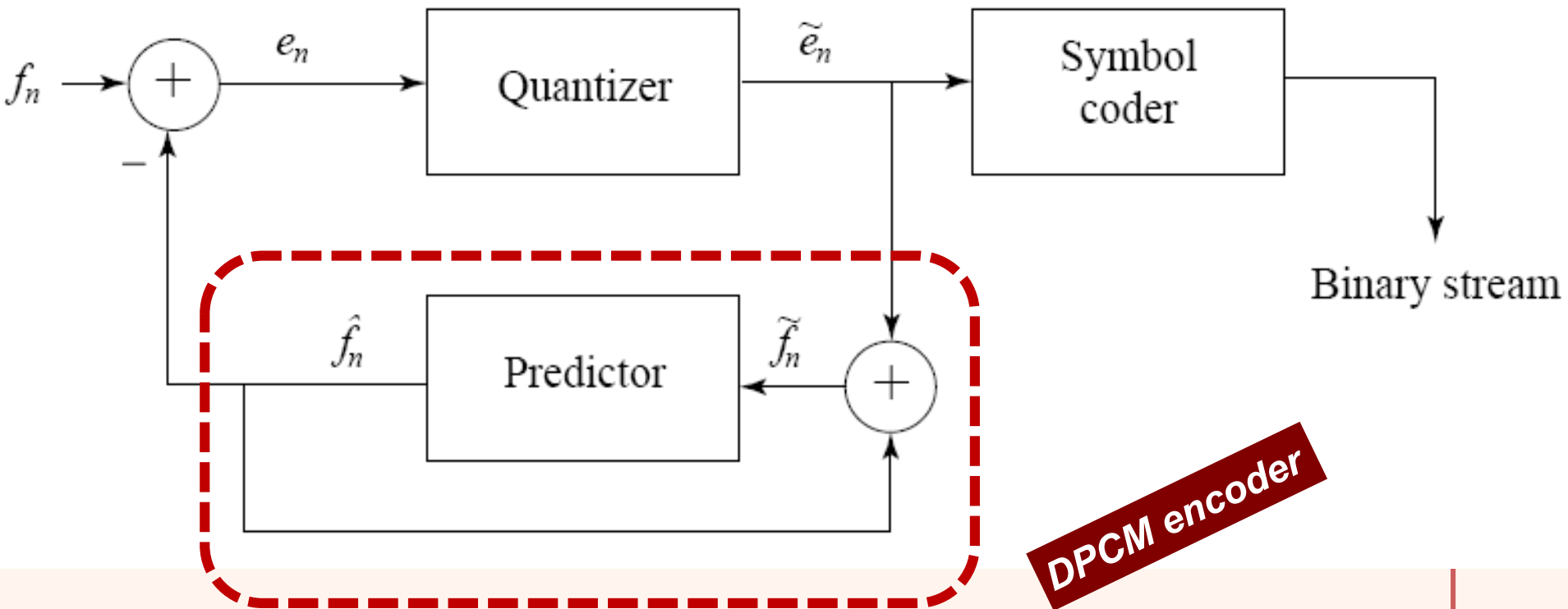
$$\begin{array}{rcccccc} e_Q(n) = & - & 1 & 4 & 0 & 0 & -1 \\ \tilde{f}(n) = & 100 & 101 & 105 & 105 & 105 & 104 \\ err(n) = & - & 1 & 15 & 15 & 15 & 14 \end{array}$$



Lossless predictive coding

$n =$	0	1	2	3	4	5
$f(n) =$	100	102	120	120	120	118
$\hat{f}(n) =$	—	100	101	105	109	113
$e(n) =$	—	2	19	15	11	5
$e_Q(n) =$	—	1	4	4	4	3
$err(n) =$	—	1	15	11	7	2





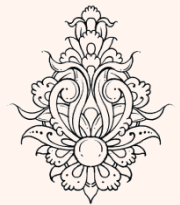
مثال

در این مثال پیش‌بینی بر اساس میانگین دو نماد قبلی صورت می‌گیرد

جدول چندی‌سازی

DPCM quantizer reconstruction levels.

e_n in range	Quantized to value
-255 .. -240	-248
-239 .. -224	-232
.	.
.	.
.	.
-31 .. -16	-24
-15 .. 0	-8
1 .. 16	8
17 .. 32	24
.	.
.	.
.	.
225 .. 240	232
241 .. 255	248



f_1	f_2	f_3	f_4	f_5
130	150	140	200	230

$$(130+154)/2$$

پیش بینی شده

$$\hat{f} = 130, 130, 142, 144, 167$$

$$e = 0, 20, -2, 56, 63$$

$$\tilde{e} = 0, 24, -8, 56, 56$$

$$\tilde{f} = 130, 154, 134, 200, 223$$

دوباره سازی شده

