

## RPC Encoder/Decoder IP

### 1. 概要

UrbanetDesign 独自フォーマットの画像圧縮/伸長 IP です。

ブロック単位(64pixel×2pixel)で圧縮後のデータサイズが一定になることを保証しますので、画像処理システム内で、処理部とメモリコントローラーの間に挿入することで、メモリ帯域の削減を行うことが可能です。

### 2. 特徴

- ・ 固定圧縮率でありながら、劣化の少ない圧縮アルゴリズムです。
- ・ ハードウェア処理向きのアルゴリズムなので、小規模の **Logic** で実現できます。
- ・ ブロック単位で完結した圧縮アルゴリズムなので、ブロック境界をまたがなければ、部分切り出し等のランダムアクセスが可能です。

### 3. 用途

各種画像処理システムの外部メモリ帯域削減など

### 4. カスタマイズ

各種カスタマイズ承ります。

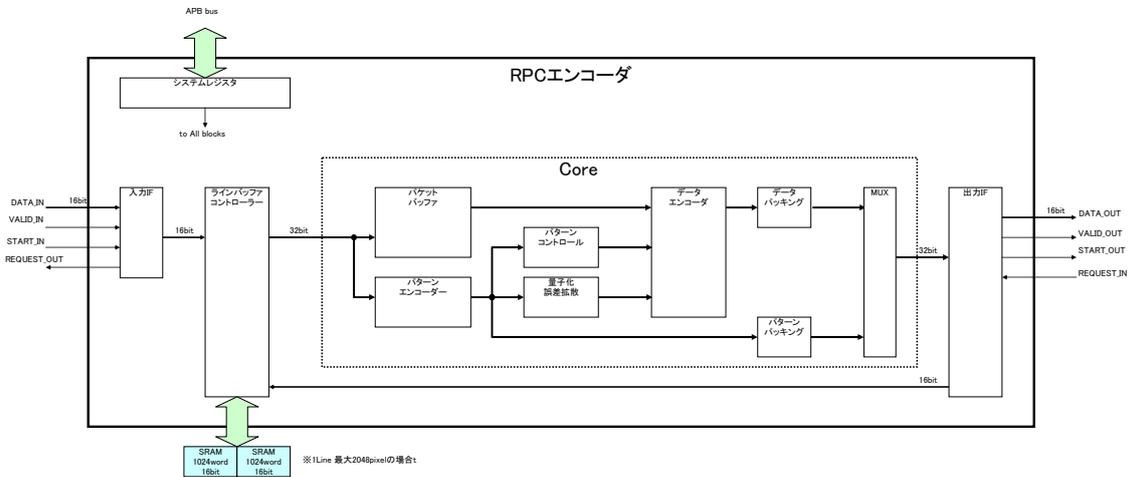
カスタマイズ例)

- ・ ブロックサイズ変更
- ・ RGB フォーマット入力
- ・ 10bit 精度対応
- ・ 2 ライン同時入力
- ・ CPU I/F 変更

## 5. Encoder 部スペック

圧縮フォーマット	UrbanetDesign 独自フォーマット(RPC 方式)
入力フォーマット	YCC 422 (Y8bit,C8bit 同時入力) ラスタ順入力 (コア部は 2 ライン同時入力の 32bit)
出力フォーマット	16bit RPC format (コア部は 2 ライン同時出力の 32bit)
最大入力サイズ	最大 4032×4094 pixel
圧縮率	固定圧縮率 (75%~46.875%の 10 段階からレジスタで選択可能)
ブロックサイズ	64pixel×2pixel
ゲート規模	約 47KGate
動作速度(参考)	200MHz (45nm LowPower ASIC), 125MHz (Altera StratixIV)
メモリ (2Line 並び替え用)	1 ライン分のシングルポート SRAM (例:入力画像サイズの幅が 1024pixel の場合 16Kbit)
処理レート	ピクセル入力 最大 16bit/CLK (Core は 32bit/CLK) データ出力 最大 16bit/CLK*圧縮率 ※ 原則、出力先のモジュールからの REQUEST_IN が常に H の場合はデータによらず 16bit/CLK を保証します。

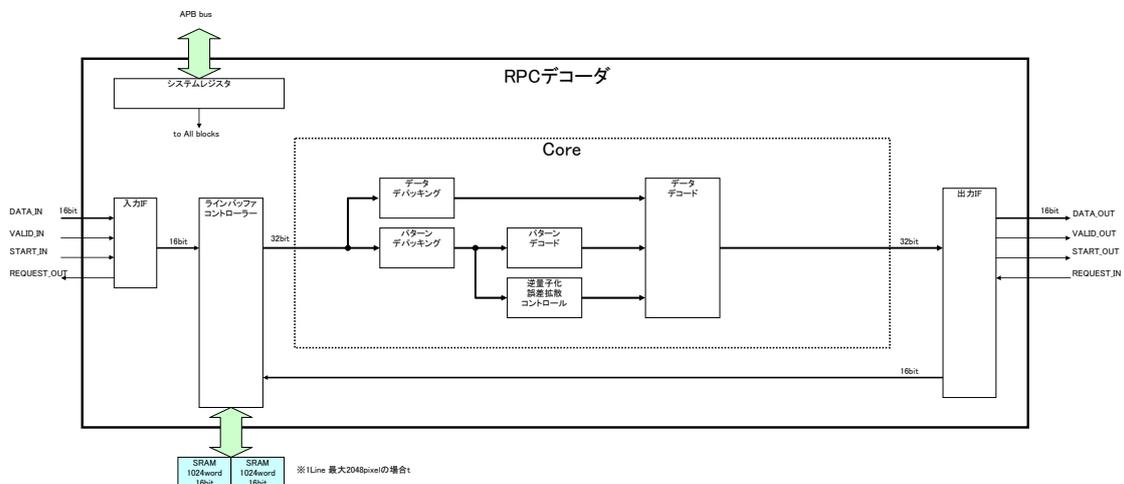
### Encoder 部ブロック構成



## 6. Decoder 部スペック

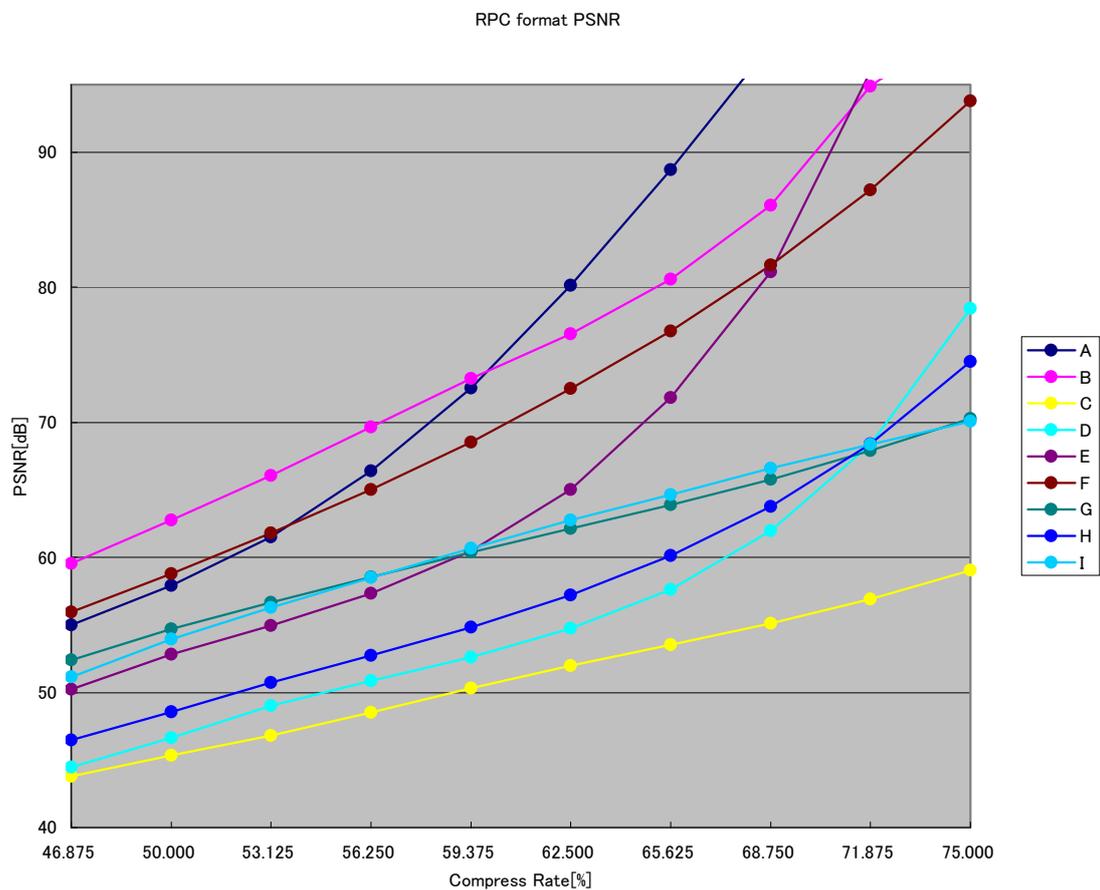
圧縮フォーマット	UrbanetDesign 独自フォーマット(RPC 方式)
入力フォーマット	16bit RPC format (コア部は 2 ライン同時出力の 32bit)
出力フォーマット	YCC 422 (Y8bit,C8bit 同時入力) ラスタ順入力 (コア部は 2 ライン同時入力の 32bit)
最大入力サイズ	最大 4032×4094 pixel
圧縮率	固定圧縮率 (75%~46.875%の 10 段階からレジスタで選択可能)
ブロックサイズ	64pixel×2pixel
ゲート規模	約 29KGate
動作速度(参考)	200MHz (45nm LowPower ASIC), 108MHz (Altera StratixIV)
メモリ (2Line 並び替え用)	1 ライン分のシングルポート SRAM (例:入力画像サイズの幅が 1024pixel の場合 16Kbit)
処理レート	データ入力 最大 16bit/CLK*圧縮率 ピクセル出力 最大 16bit/CLK (Core は 32bit/CLK) ※原則、入力元のモジュールからの VALID_IN が常に H の場合はデータによらず 16bit/CLK を保証します。

### Decoder 部ブロック構成



## 7. 圧縮画質

RPC フォーマットでの各圧縮率での画質(PSNR)は下記のようになっております。

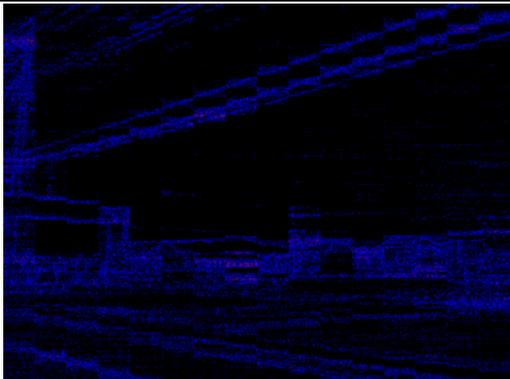
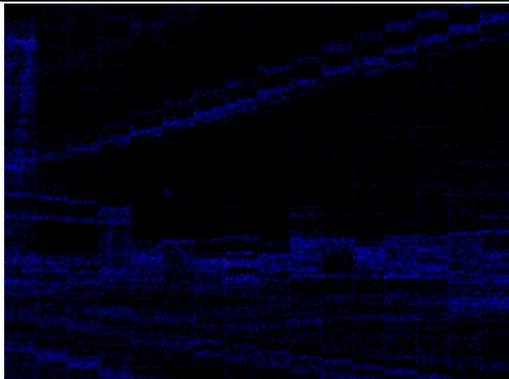


		
<p>画像 A 1024×768 pixel</p>	<p>画像 B 1024×768 pixel</p>	<p>画像 C 1024×768 pixel</p>
		
<p>画像 D 1024×768 pixel</p>	<p>画像 E 1024×768 pixel</p>	<p>画像 F 2048×1600 pixel</p>
		
<p>画像 G 768×1024 pixel</p>	<p>画像 H 768×1024 pixel</p>	<p>画像 I 640×960 pixel</p>

入力画像と、エンコード→デコード処理を行なった出力画像の差分は下記のようになります。

画像 A

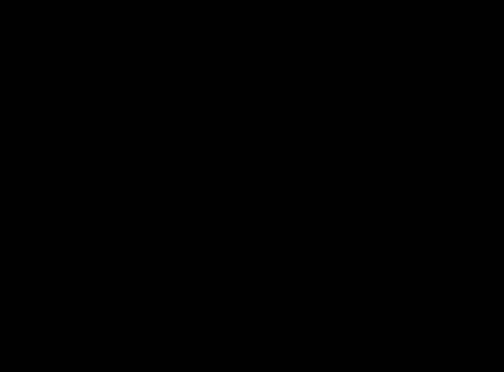
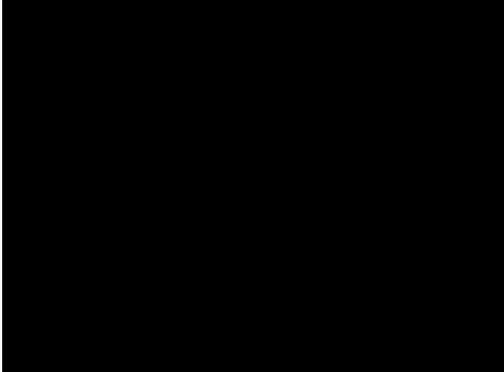
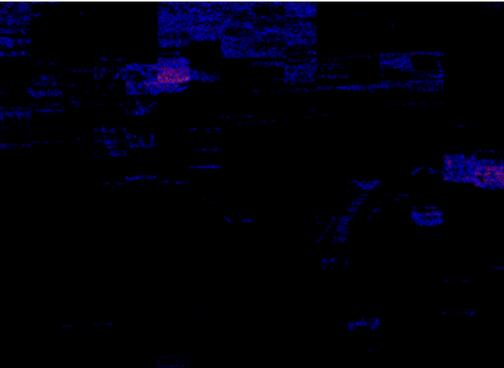


75%圧縮時 Y 成分	75%圧縮時 C 成分
	
50%圧縮時 Y 成分	50%圧縮時 C 成分
	

黒: 差分 0  
青: 差分±1  
赤: 差分±2  
マゼンタ: 差分±3  
緑: 差分±4  
シアン: 差分±5  
黄色: 差分±6  
白: 差分±7 以上

画像 B



75%圧縮時 Y 成分	75%圧縮時 C 成分
	
50%圧縮時 Y 成分	50%圧縮時 C 成分
	

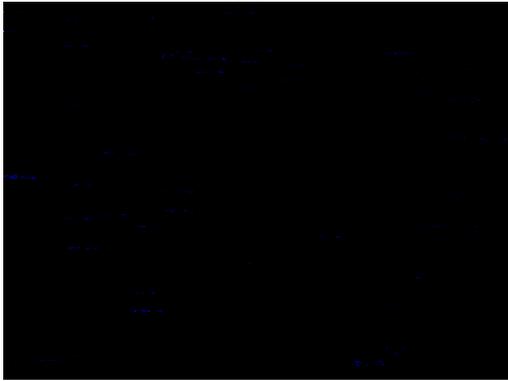
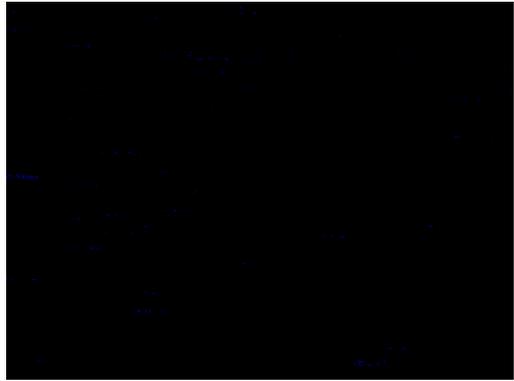
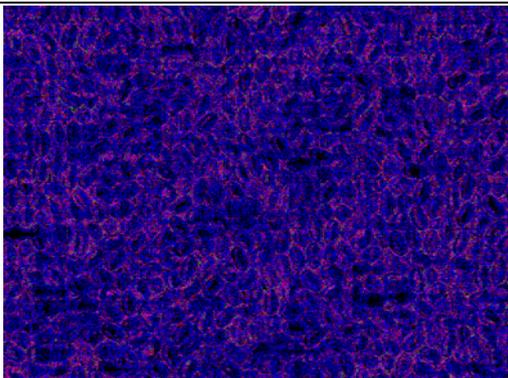
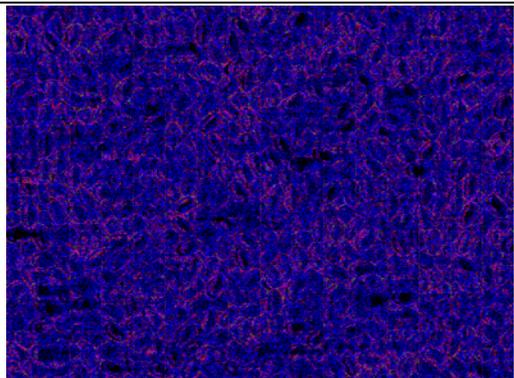
画像 C



75%圧縮時 Y 成分	75%圧縮時 C 成分
A visualization of the Y component of the image after 75% compression. The image is almost entirely black, with only a few scattered blue pixels representing the original tree's structure.	A visualization of the C component of the image after 75% compression. The image is almost entirely black, with only a few scattered blue pixels representing the original tree's structure.
50%圧縮時 Y 成分	50%圧縮時 C 成分
A visualization of the Y component of the image after 50% compression. The image shows more detail, with blue and purple pixels forming the outline and some internal structure of the tree.	A visualization of the C component of the image after 50% compression. The image shows more detail, with blue and purple pixels forming the outline and some internal structure of the tree.

画像 D



75%圧縮時 Y 成分	75%圧縮時 C 成分
	
50%圧縮時 Y 成分	50%圧縮時 C 成分
	

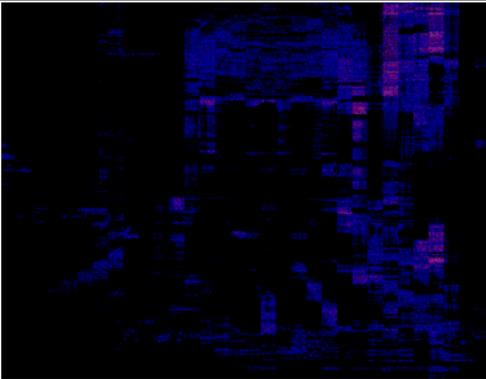
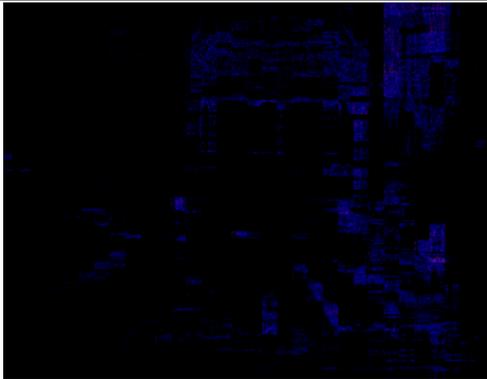
画像 E



75%圧縮時 Y 成分	75%圧縮時 C 成分
A square image that is almost entirely black, representing the Y component of the image after 75% compression. Only a few very faint, scattered pixels are visible.	A square image that is almost entirely black, representing the C component of the image after 75% compression. Only a few very faint, scattered pixels are visible.
50%圧縮時 Y 成分	50%圧縮時 C 成分
A square image showing a noisy, blue-tinted representation of the Y component after 50% compression. Some faint, dark shapes corresponding to the child and flowers are visible through the noise.	A square image showing a noisy, blue-tinted representation of the C component after 50% compression. Some faint, dark shapes corresponding to the child and flowers are visible through the noise.

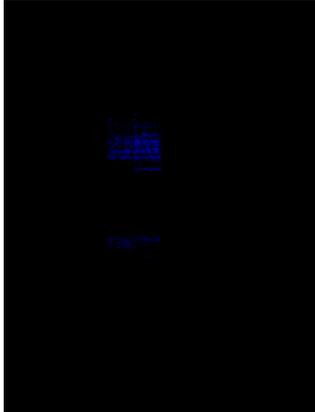
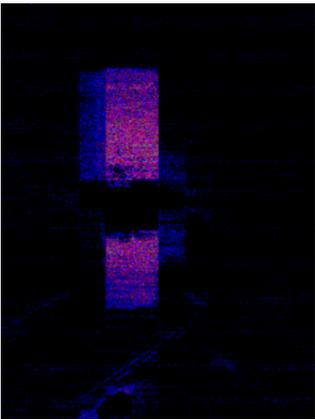
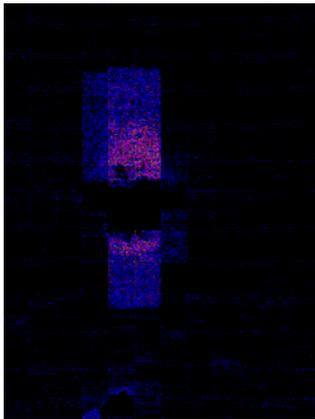
画像 F



75%圧縮時 Y 成分	75%圧縮時 C 成分
	
50%圧縮時 Y 成分	50%圧縮時 C 成分
	

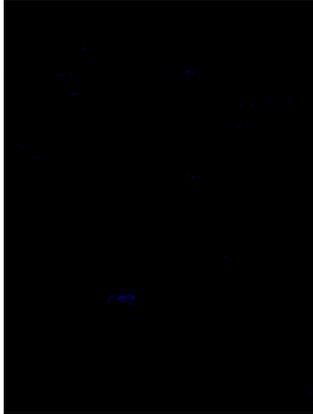
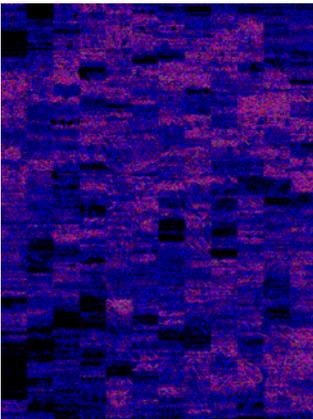
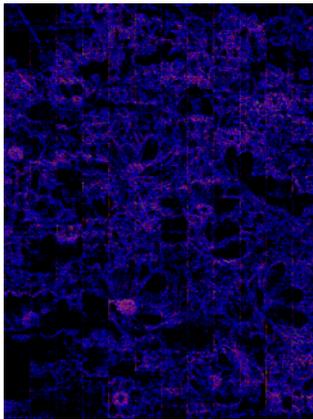
画像 G



75%圧縮時 Y 成分	75%圧縮時 C 成分
 A visualization of the Y component at 75% compression. It shows a dark background with a small, bright blue and purple rectangular area in the center, representing the lamp's position.	 A visualization of the C component at 75% compression. It shows a dark background with a small, bright blue and purple rectangular area in the center, representing the lamp's position.
50%圧縮時 Y 成分	50%圧縮時 C 成分
 A visualization of the Y component at 50% compression. It shows a dark background with a larger, bright blue and purple rectangular area in the center, representing the lamp's position.	 A visualization of the C component at 50% compression. It shows a dark background with a larger, bright blue and purple rectangular area in the center, representing the lamp's position.

画像 H



75%圧縮時 Y 成分	75%圧縮時 C 成分
 A visualization of the Y component of the image after 75% compression. The image is almost entirely black, with only a few faint, scattered blue and purple pixels visible, indicating significant loss of color and detail.	 A visualization of the C component of the image after 75% compression. The image is almost entirely black, with only a few faint, scattered blue and purple pixels visible, indicating significant loss of color and detail.
50%圧縮時 Y 成分	50%圧縮時 C 成分
 A visualization of the Y component of the image after 50% compression. The image shows a dense, noisy pattern of blue and purple pixels, with some darker areas, indicating significant loss of color and detail.	 A visualization of the C component of the image after 50% compression. The image shows a dense, noisy pattern of blue and purple pixels, with some darker areas, indicating significant loss of color and detail.

画像1



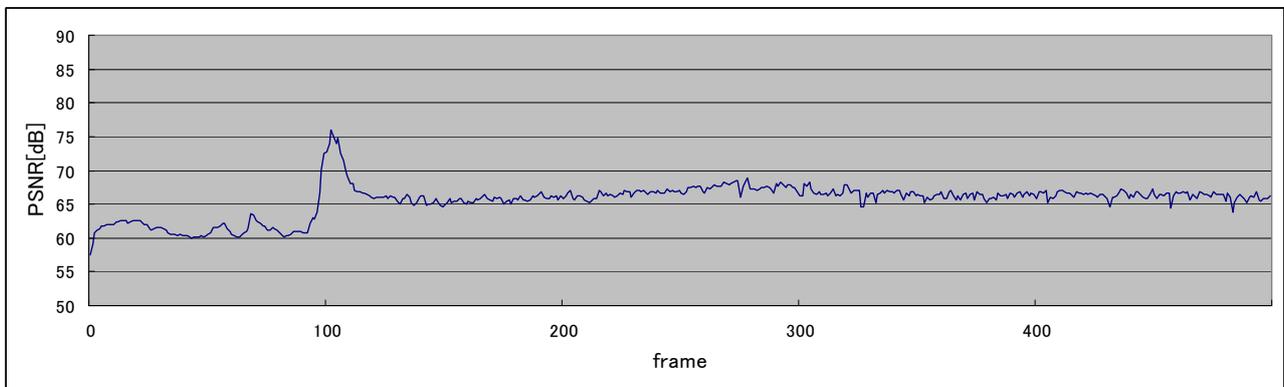
75%圧縮時 Y 成分	75%圧縮時 C 成分
50%圧縮時 Y 成分	50%圧縮時 C 成分

# UrbaNetDesign

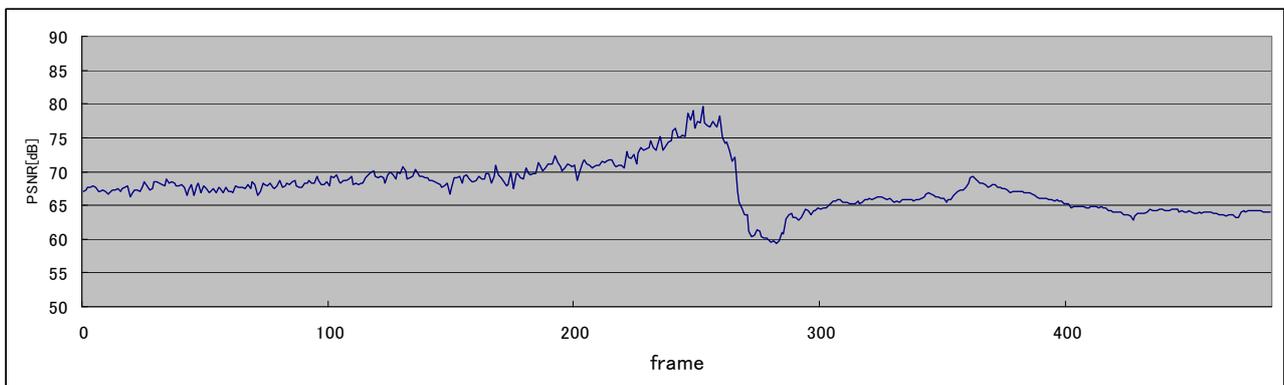
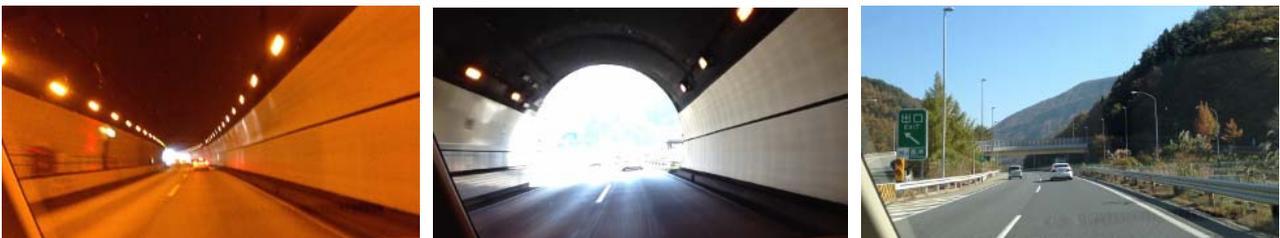
## 動画への実施例

50%圧縮モードにて、fullHD 動画に使用した場合の PSNR の変化を以下に示します。

### ・動画 T0

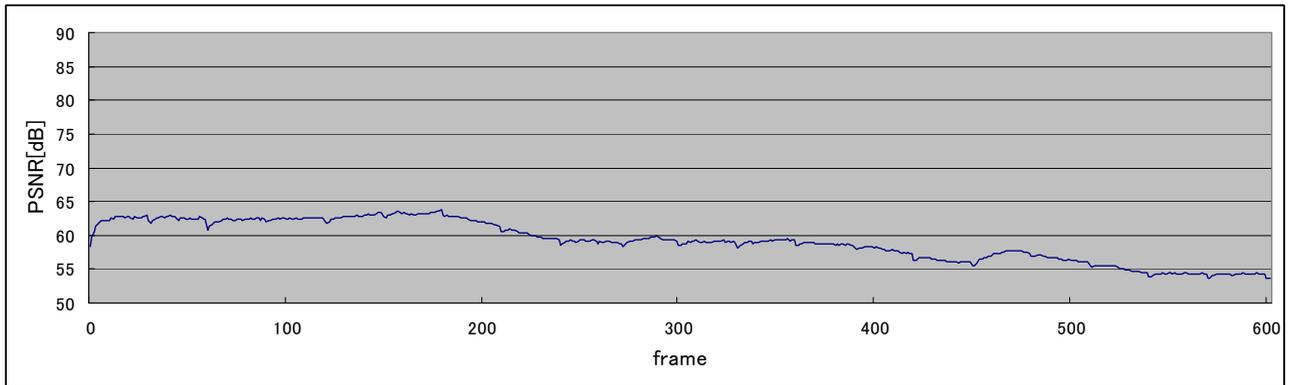


### ・動画 T1

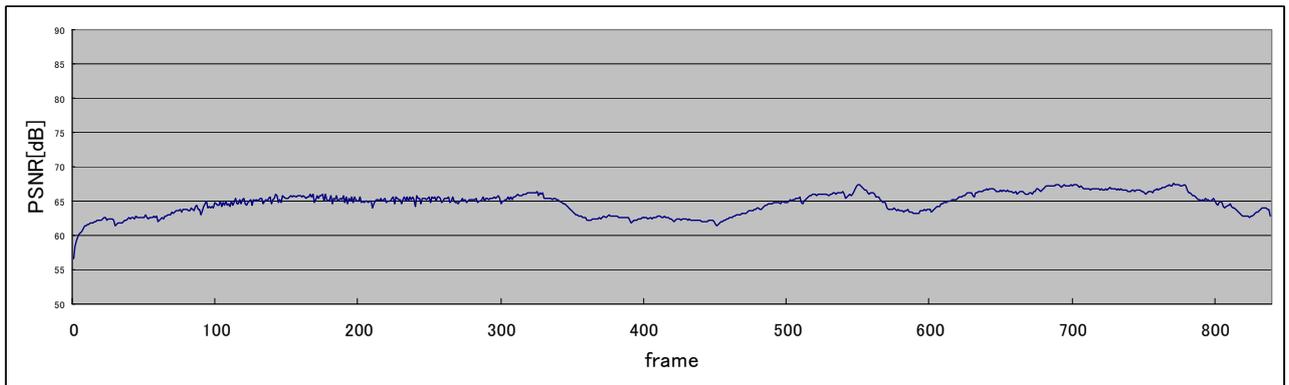


# UrbaNetDesign

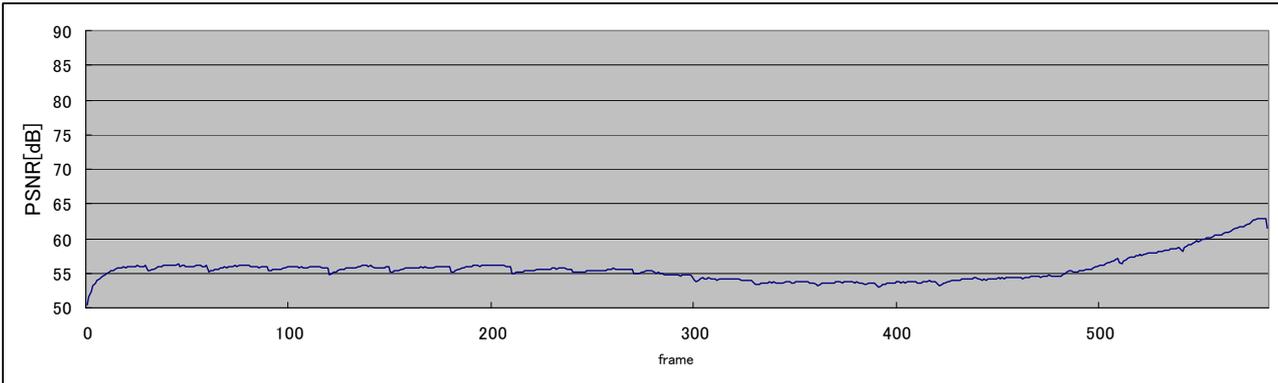
## ・動画 K0



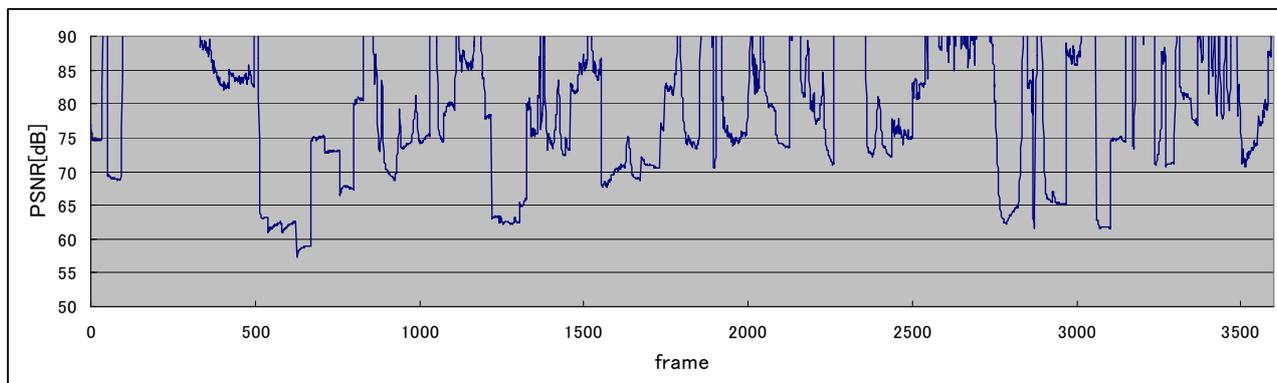
## ・動画 K1



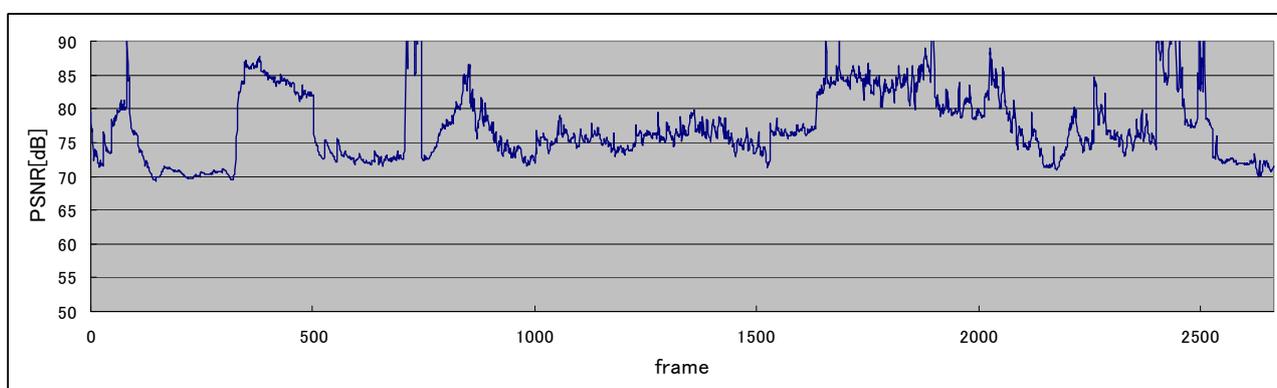
• 動画 K2



- アニメーション作品プロモーション映像



- アイドルグループ ライブ映像



- 劇場版映画のブルーレイ版プロモーション映像

