

チョン ジーン

国立情報学研究所、コンテンツ科学研究系

2013年11月26日

どこでもビジュアルコミュニケーション ～ 逆境を克服する画像処理 ～

Acknowledgement

共同研究者:

- Y. Feng, Y. Mao, Z. Liu, 計字生教授 (国立情報学研究所, 日本)
- 久保田彰准教授 (中央大学, 日本)
- W. Sun, W. Hu, P. Wan, O. Au教授 (香港科技大学, 香港)
- W. Kim, A. Ortega教授 (南カリフォルニア大学, アメリカ)
- D. Florencio, C. Zhang, P. Chou (マイクロソフト研究所、アメリカ)
- H. Hadizadeh, Y. Gao, J. Liang准教授, I. Bajic准教授 (サイモンフレーザー大学、カナダ)
- B. Machiavello, C. Dorea, M. Hung (ブラジリア大学、ブラジル)
- W.-t. Tan (ヒューレットパッカード研究所、アメリカ)
- 菅野勝、三功ひろし (KDDI研究所、日本)

プレゼンテーション アウトライン

- 背景
- ビジュアルコミュニケーションとは？
- 2次元動画から3次元動画へ
- スパース表現
 - 変換符号化
 - グラフ変換
 - **応用**: 奥行き画像の圧縮、ノイズ除去、画像補間
- 顕著性マップ
 - 定義、例
 - **応用**: エラー隠蔽、画像修復

経歴

- 2000年にカリフォルニア大学バークレー校に工学大学院電子工学部で博士を取得

- 画像や動画の符号化と伝送

2次元動画

コミュニケーション

(12 yrs)

- 2000年から2009年まで日本ヒューレットパカード研究所で主任研究員

- 動画の圧縮とストリーミング

- マルチビュー動画の符号化

3次元動画

コミュニケーション

(7 yrs)

- 2009年11月から国立情報学研究所に所属

- 自由視点画像、動画処理

- 臨場感ある (immersive) ビジュアルコミュニケーション



背景



送信側

- **背景:** ビジュアルセンサーは安い。
- **目標:** ネットワークを通して、どこの**送信側**からビジュアルメディアをどこの**受信側**まで**リアルタイム**で送信→**ビジュアルコミュニケーション**。

ネットワーク

- **アプリケーション:**

- **テレビ:** HD, 4k, 8k, 自由視点。
- **オンデマンドネットワークストリーミング:** YouTube, Hulu。
- **テレビ会議:** Skype, 3D, 臨場感がある。
- **テレメディスン:** 医者が遠距離で診察。
- **遠隔教育:** 講義の放送。
- **監視カメラ:** ナイトビジョン。

受信側

ビジュアルコミュニケーションの壁 (2D動画)



- **送信側:**

1. ビジュアルデータの取得(ノイズ除去)。
2. ビジュアルデータの符号化。



- **ネットワーク:**

1. ストリーミング動画用のパケットの締切。
2. ネットワーク帯域幅の限界。
3. パケットの損失と遅延。



- **受信側:**

1. 無くしたビジュアルデータの隠ぺい。

2次元から3次元動画へ！

動画コミュニケーション: 2D, 2.5D, 3D

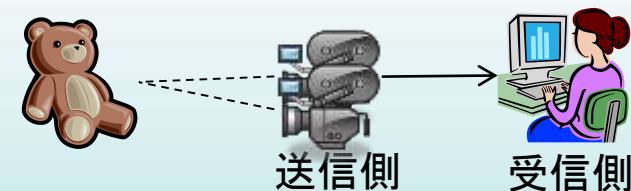
• 2D 動画

- 送信側にカメラ一台。
- 受信側に2Dモニター。(インタラクション無し)



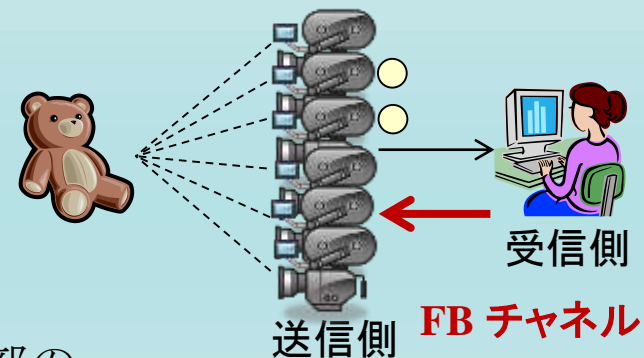
• 2.5D 動画 (ステレオ動画)

- 送信側にカメラ二台。
- 受信側にステレオモニター。(インタラクション無し)



• 3D 動画 (マルチビュー、自由視点)

- 送信側にカメラ複数台。
- 受信側にモニター。(インタラクションあり)
 - 送信側が多次元のメディアを撮影、受信側が一部のみ観察。



カラー画像



奥行き画像

マルチビュー動画

- 固定しているカメラからビューを選択。



自由視点（free viewpoint）動画

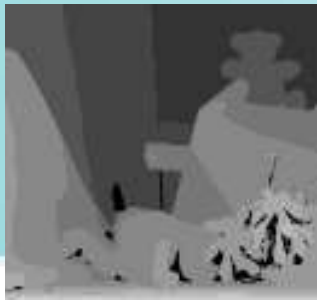
- 仮想視点から画像を合成し、鑑賞する。



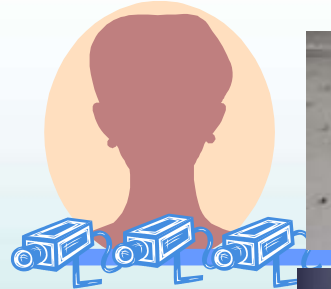
ビジュアルコミュニケーションの壁 (3D動画)



カラー画像



奥行き画像



送信側:

1. 奥行きデータの取得 (ノイズ除去, 超解像)
2. 奥行きデータの符号化



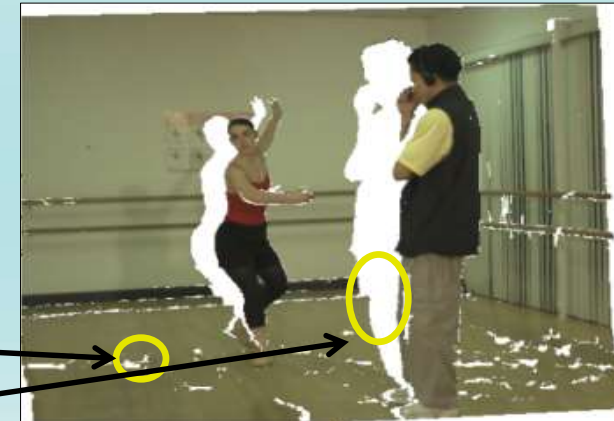
カメラ視点からの画像

ネットワーク:

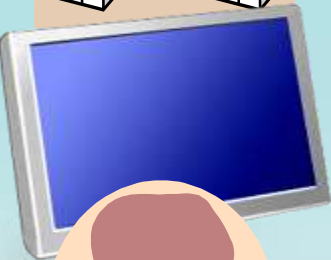
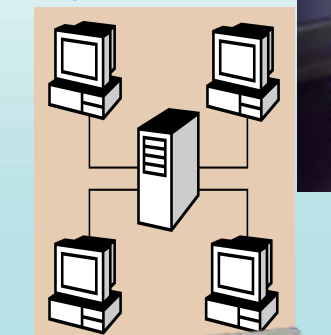
...

受信側:

1. 仮想視点からの画像合成
 1. 画像補間 (interpolation)
 2. 画像修復 (inpainting)



仮想視点からの画像



今日検討するビジュアルコミュニケーション課題一覧



カラー画像

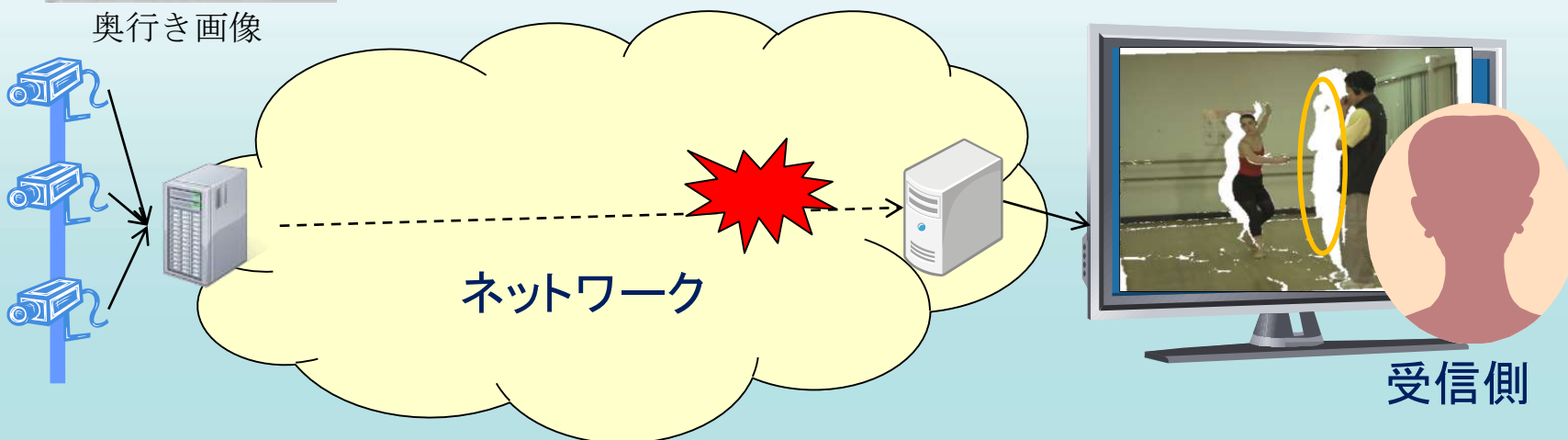


奥行き画像

2. パケット損失による
エラー隠蔽



送信側



受信側

1. 奥行き画像の符号化
とノイズ除去

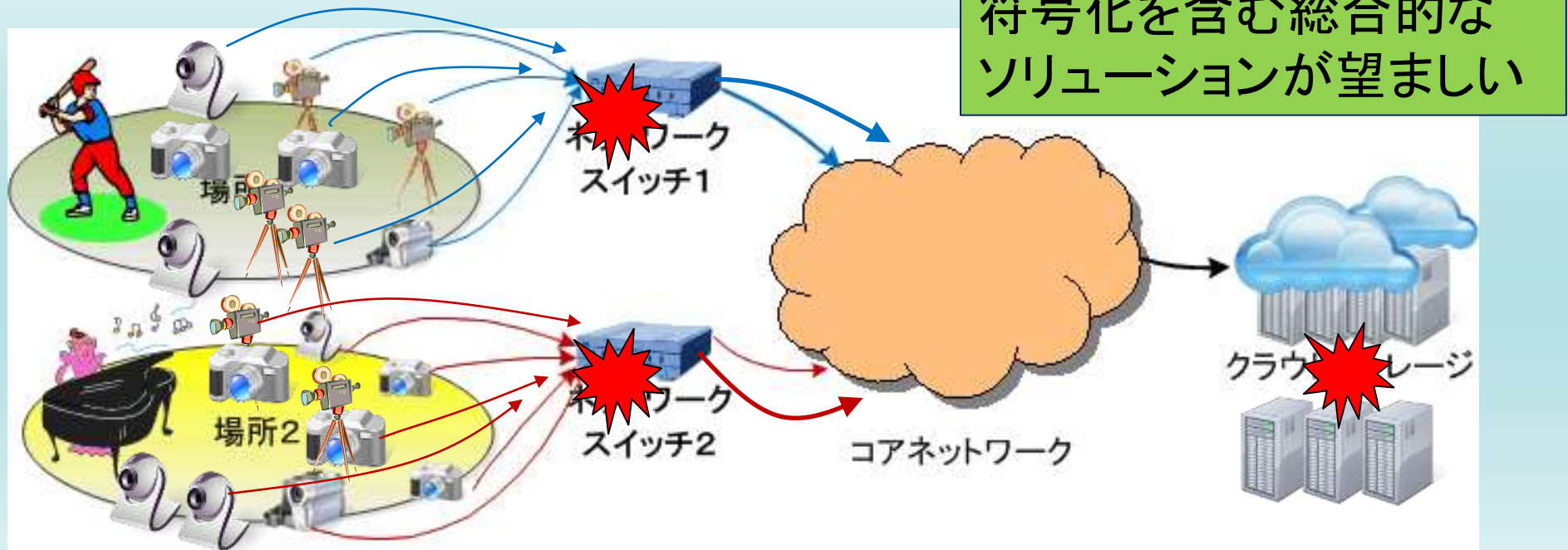
3. 画像修復、補間

プレゼンテーション アウトライン

- 背景
- ビジュアルコミュニケーションとは？
- 2次元動画から3次元動画へ
- スパース表現
 - 変換符号化
 - グラフ変換
 - **応用:** 奥行き画像の圧縮、ノイズ除去、画像補間
- 顕著性マップ
 - 定義、例
 - **応用:** エラー隠蔽、画像修復

ビッグデータ時代の動画符号化

- **質問:** 動画符号化まだ必要？
- データ量が年々 60% 増えている。
- 「データ量の増加スピード」は、「ネットワーク帯域幅拡大スピード」と「ストレージの減額スピード」より速い。



変換符号化 (Transform Coding)

- **JPEG** (静止画像を圧縮する方式)

- 8x8 ピクセルブロックに分ける。
- 離散コサイン変換、Discrete Cosine Transform (DCT)。
- 変換係数を符号化する。



$$\mathbf{a} = \Phi \mathbf{x}$$

ピクセルブロック

変換係数

DCT変換行列

$$\mathbf{x} = \Phi^{-1} \mathbf{a}$$

質問: なぜ変換する?

答え: スパース表現.

$$\mathbf{a} = \begin{bmatrix} a_0 \\ a_1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

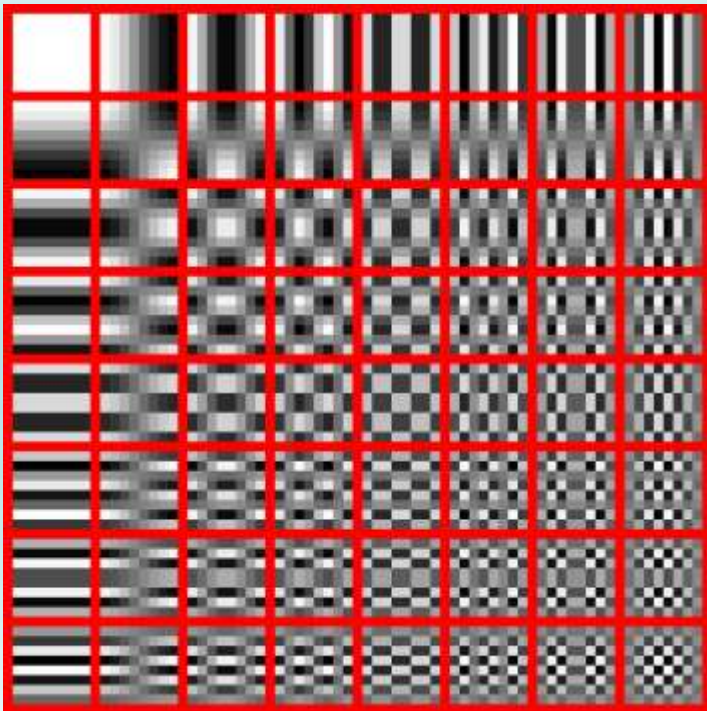
← 係数が殆ど0.

離散コサイン変換 (DCT)

- 離散コサイン変換 (DCT) type 2:

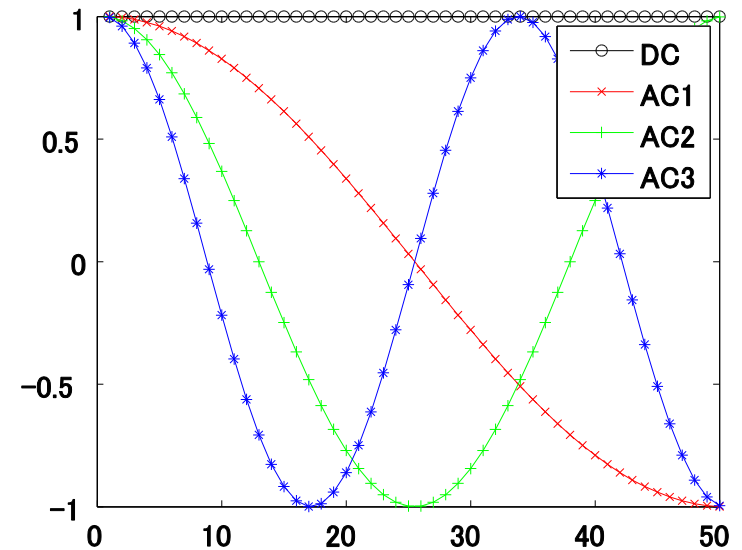
$$k = 0, \dots, N-1 \quad X_k = \sum_{n=0}^{N-1} x_n \cos\left(\frac{\pi}{N}\left(n + \frac{1}{2}\right)k\right)$$

2D DCT basis is set of outer-product of 1D DCT basis in x- and y-dimension.



$$\mathbf{a} = \Phi \mathbf{x}$$

$$\mathbf{a} = \begin{bmatrix} a_0 \\ a_1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$



$$a_k = \varphi_k^T \mathbf{x}$$



一般的な画像から取ったピクセルブロックは、高い周波数は殆どない。



スパース表現 (Sparse Representation)

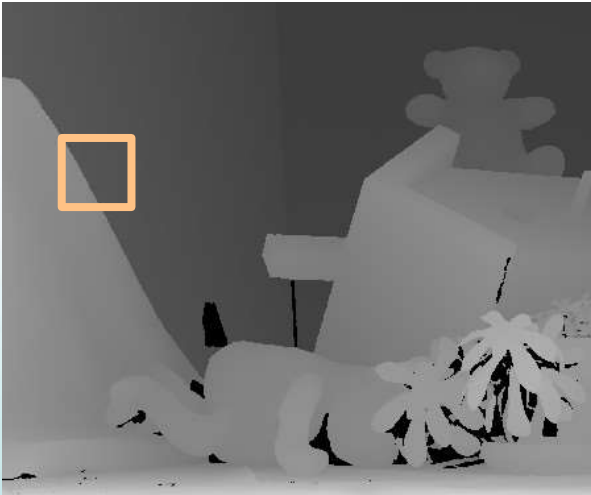
- “Among competing hypotheses, the hypothesis with the fewest assumptions should be selected.” **Occam's razor**
- 「ある事柄を説明するためには、必要以上に多くを仮定するべきでない」 **オッカムの剃刀**
- “Everything should be as simple as possible, but not simpler.” **Albert Einstein**
- 「物事は全て、出来る限り単純にすべきだ。」 **アインシュタイン**

$$\mathbf{a} = \Phi \mathbf{x} \quad \longleftrightarrow \quad \mathbf{x} = \Phi^{-1} \mathbf{a}$$

変換係数 変換行列 信号

質問: 一番適切な変換行列は？

グラフ変換 Graph Transform (GT)



- **観測:**

- ピクセルブロック内に境界線があると、DCT変換係数はスパースではない。

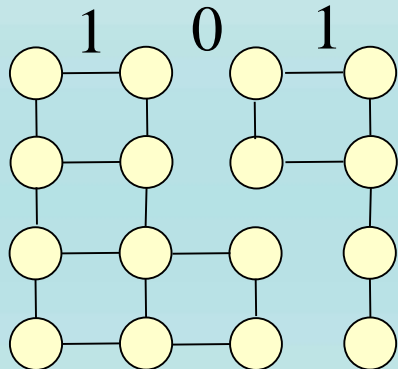
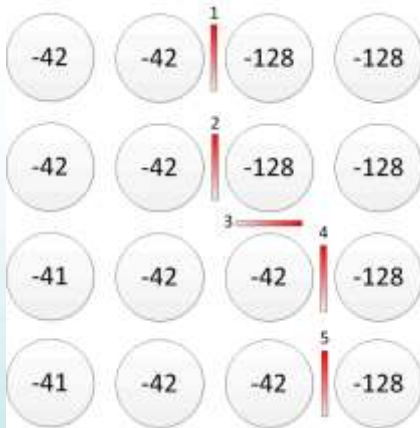
- **アイデア** :*

- **Graph Transform (GT)**によるスパース表現

1. DCT変換は固定的。GT変換は信号に適応的。
2. ピクセル間の関連性をグラフで表現する。



GTの作成



ステップ1:

- ピクセルの間の関連性を測る。

ステップ2:

- ピクセルブロックからグラフを作成する。

ステップ3:

- グラフからグラフ変換行列(GT)を計算する。

1. **Graph Laplacian*** 行列を定義する $L = D - A$.
2. L から固有ベクトル (eigenvector) を求める。

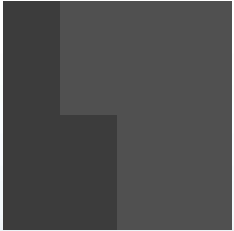
degree行列

adjacency行列

Intuition:

似ているサブグラフに分ける → スパース行列係数

GTによる奥行き画像の符号化



DCT

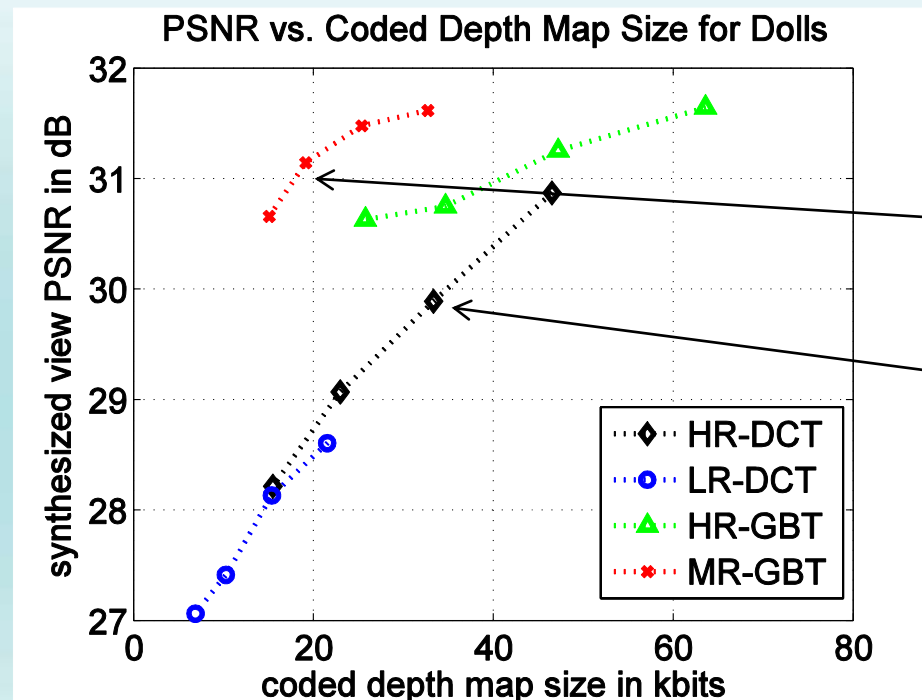
$$\alpha_2 = \begin{bmatrix} 285 & -29 & -5 & -4 \\ 16 & 1 & -16 & -4 \\ -5 & 3 & 5 & -7 \\ -1 & -4 & 1 & 9 \end{bmatrix}$$

GT

$$\alpha_1 = \begin{bmatrix} 237 & 0 & 0 & 0 \\ 163 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

実験:

- Middlebury テストシケンス Dolls.
- Dolls: DCTより6.5%ビットレート削減。



GT

DCT

GTによる奥行き画像の符号化

DCT

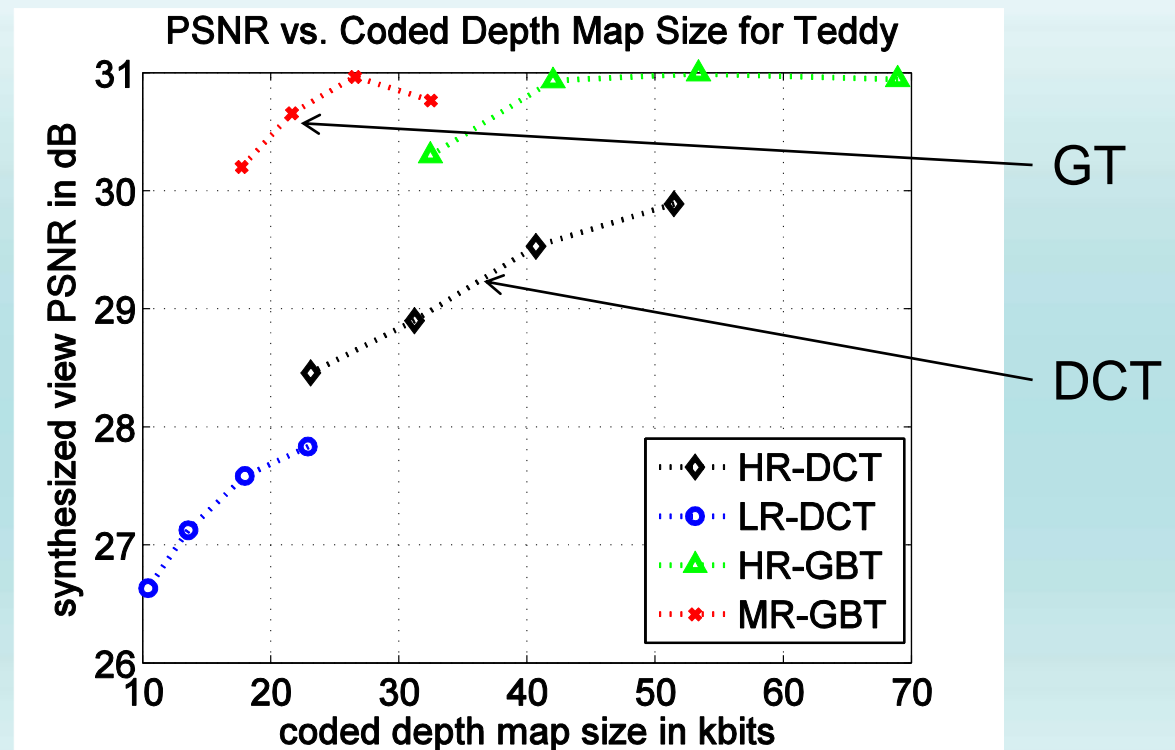


GT



実験:

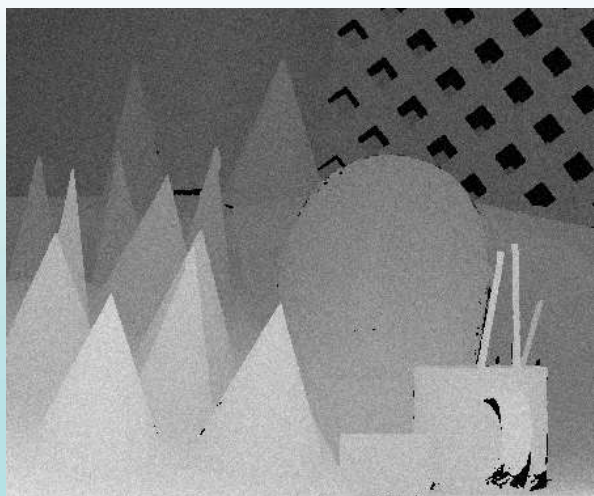
- Middlebury テストシケンス Teddy.
- Teddy: DCTより6-8%ビットレート削減。



グラフ変換によるノイズ除去

- **問題:**

- 取得した奥行き画像はnoisy。



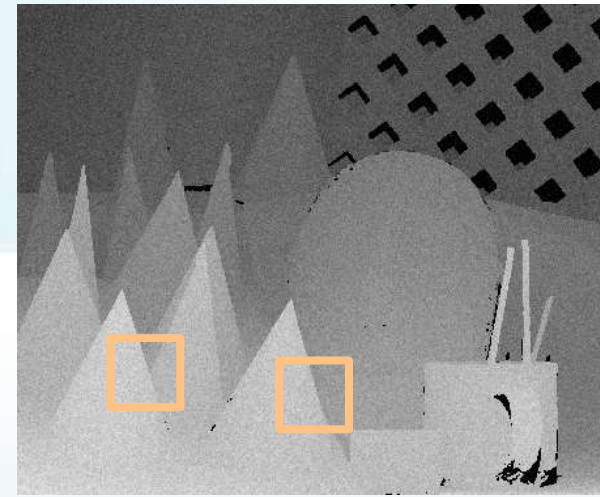
- **アイデア:***

- **Graph Transform (GT)**によるスパース表現.
 1. 適切なGT変換を求める。
 2. GT領域でスパース表現を求める。

Intuition:

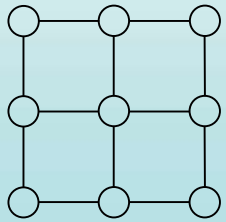
スパース表現できる信号 = ノイズを除いた信号

ノイズ除去アルゴリズム



手法:

1. 似ているピクセルパッチを探し出し、平均パッチを計算する。（自己相似）
2. 平均パッチを用いて、ピクセル間の関連性を計算する。グラフを作成する。
3. グラフ変換を計算する。
4. グラフ変換を用いて、スパースソリューションを求める。



$$W = [w_{ij}],$$

$$w_{ij} = e^{\frac{-\|y_i - y_j\|^2}{\sigma_w^2}}$$

$$\mathcal{L} = D - W$$

$$\mathcal{L}U = U\Lambda$$

奥行き画像のノイズ除去

- **実験設定**

- Test Middlebury depth maps: Sawtooth
- Additive White Gaussian Noise (AWGN)を加える。
- Bilateral Filtering (BF), Non-Local Means Denoising, (NLM), Block-Matching 3D (BM3D) と比べる。

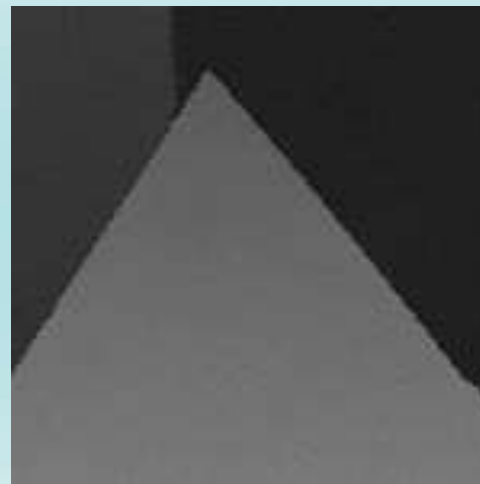
- **結果:** BM3Dより、2.28dB改善した。



NLGBT



BM3D



NLM



BF

奥行き画像のノイズ除去

- **実験設定**

- テストシケンス: Middlebury's Sawtooth
- ノイズを除去した画像を用いて、イメージレンダリングする。

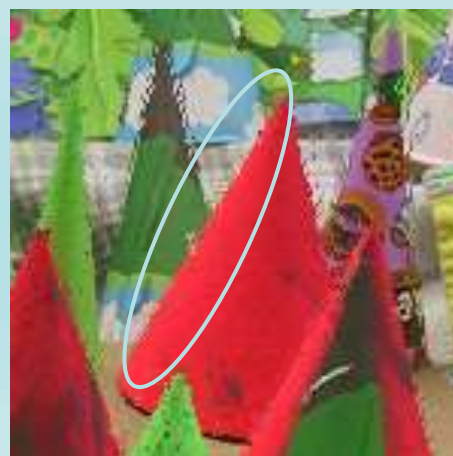
- **結果:** BM3Dより、0.60dB改善した。



NLGBT



BM3D



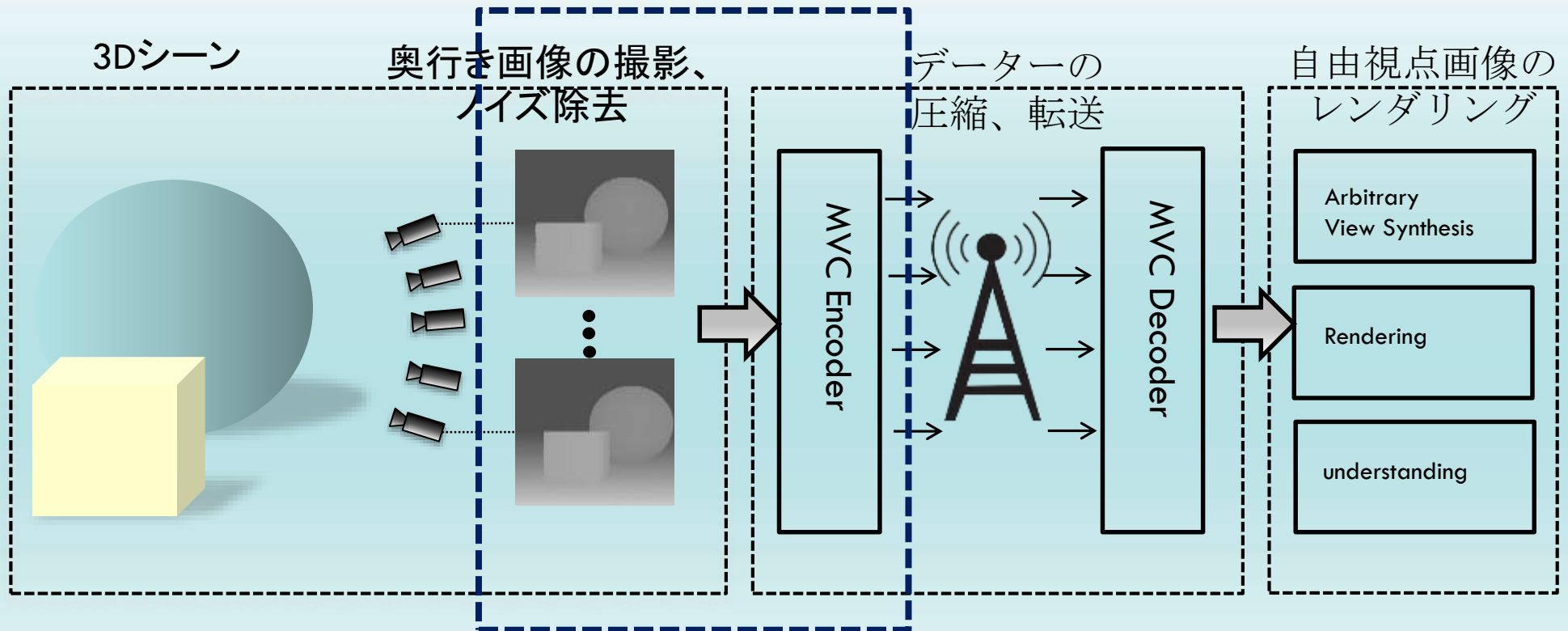
NLM



BF

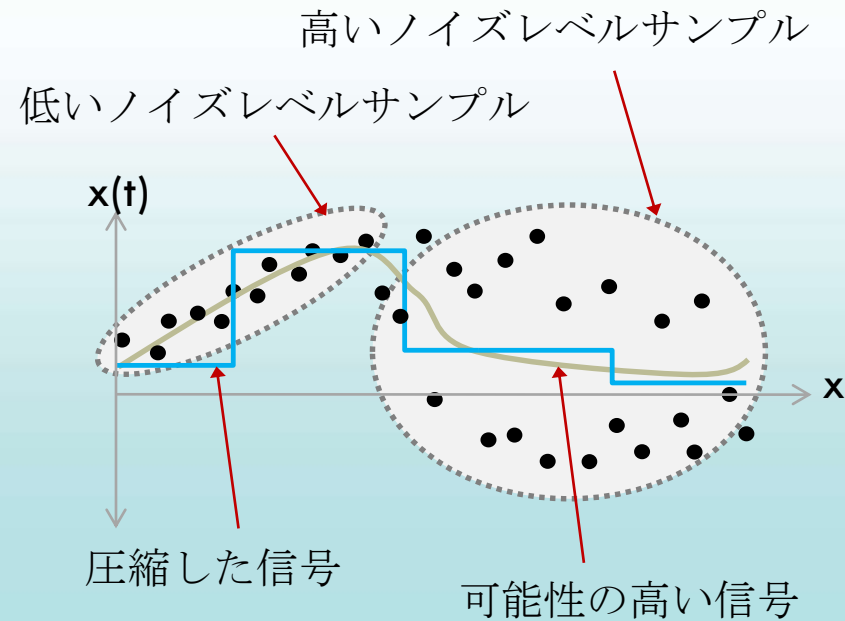
3D動画コミュニケーション

- 送信側から受信側まで含むプロセス:
 - 奥行き画像のノイズ除去。
 - 奥行き画像の符号化。



奥行き画像のノイズ除去と符号化

- **二つプロセスを別々で行う:**
 1. 観察した信号から、可能性の高い信号を計算する。
 2. 可能性の高い信号から表現サイズ小さい信号に符号化する。
- **二つプロセスを同時に行う:**
 - 観察した信号から、表現サイズ小さくて可能性の高い信号を計算する。



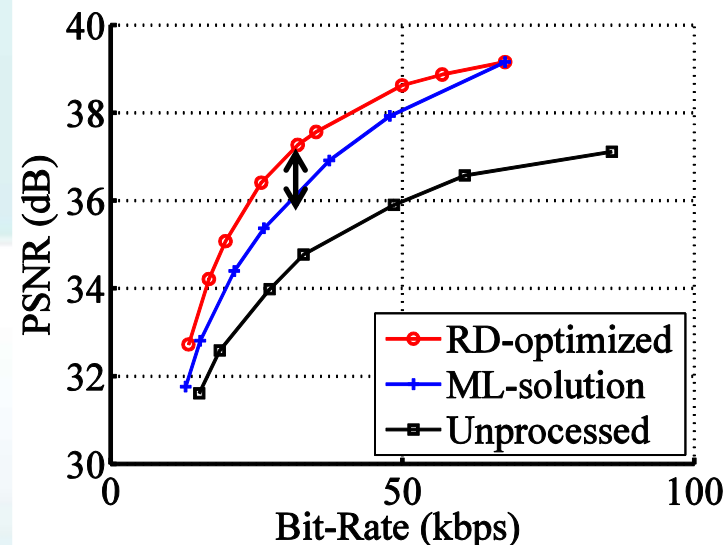
奥行き画像のノイズ除去と符号化

- **実験:**

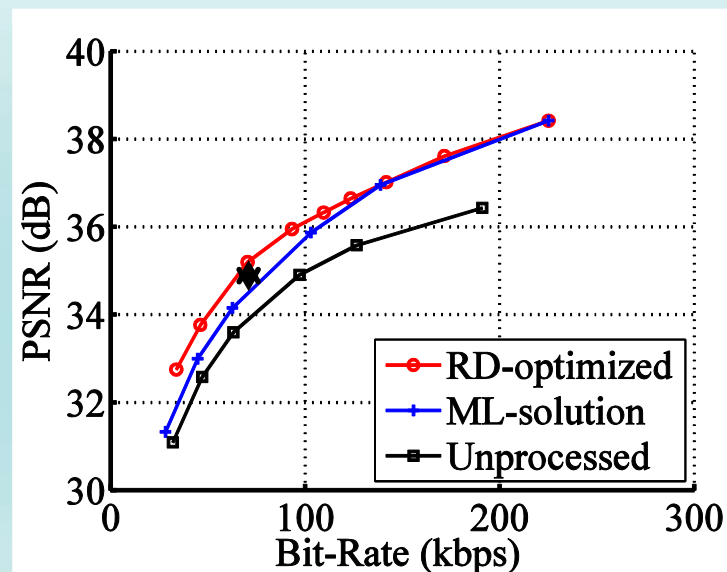
1. テストシケンス Lovebird1, Balloons
2. 符号化ツールはMVC

- **結果:**

1. Lovebird1は1.74dB改善した。
2. Balloonは0.87dB改善した。

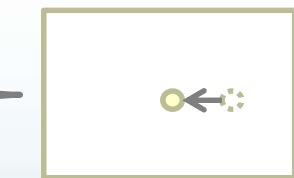
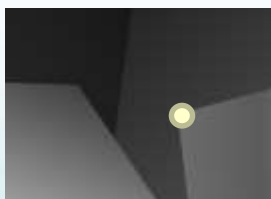


Lovebird1



Balloons

グラフ変換による画像補間

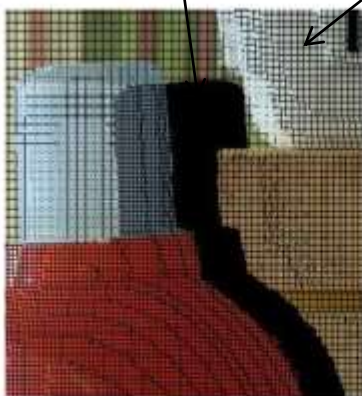


仮想視点からの画像

カメラ視点からの画像

disocclusion holes

expansion holes



(a) expansion holes



Intuition:

スパース表現できる信号 = 補間された信号

問題:

- 合成した画像(depth-image-based rendering (DIBR)に穴がある。

アイデア:*

- カメラ視点で見えるエリアに **Graph Transform (GT)** による画像補間。
 - 適切なGT変換を求める。
 - GT領域でスパース表現を求める。

グラフ変換による画像補間

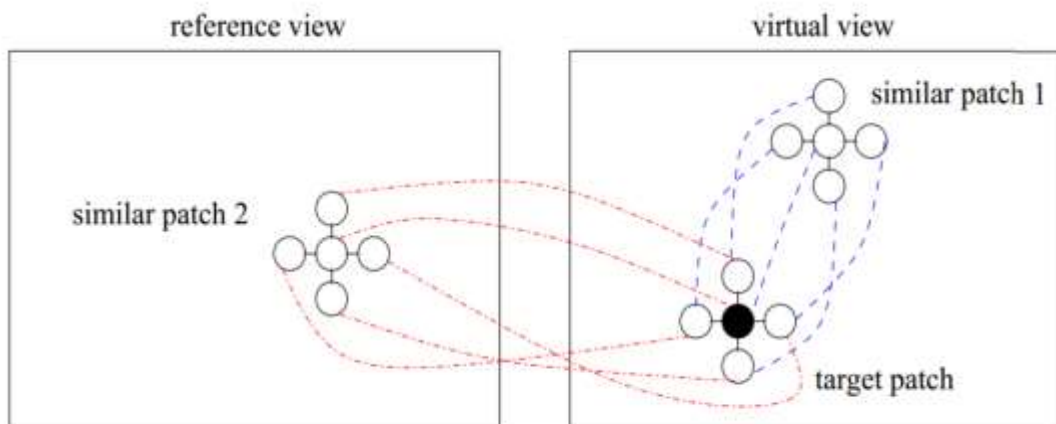


Fig. 3. Connecting pixels of similar patches to target patch

手法:

1. 似ているピクセルパッチを探し出し、リンクでパッチを繋ぐ。(自己相似)
2. リンクウェイトを計る。
3. グラフ変換領域でスパース表現を求める。



Edge weights

$$e_{i,j} = w_{i,j} u_{i,j} v_{i,j}$$

Sparse signal recovery

$$\min_{\mathbf{w}} \left\| \sum_{i=1}^N \mathbf{u}_i^T \Phi \mathbf{w} - s_i \right\|_1 + \lambda \|\mathbf{w}\|_1$$

PSNR COMPARISON FOR EXPANSION HOLE FILLING.

method	VSRs+	GBT	NLGBT
art PSNR(dB)	19.56	23.36	23.58
moebius PSNR(dB)	19.47	23.15	23.33

プレゼンテーション アウトライン

- 背景
- ビジュアルコミュニケーションとは？
- 2次元動画から3次元動画へ
- スパース表現
 - 変換符号化
 - グラフ変換
 - **応用:** 奥行き画像の圧縮、ノイズ除去、画像補間
- 顕著性マップ
 - 定義、例
 - **応用:** エラー隠蔽、画像修復

顕著性マップ (Saliency Map)



画像 1



顕著性マップ# 1

- 顕著性マップ
 - 人間の見るところ (Region of Interest) を予測する。
- 計算方式
 - ローレベルの特徴 (色や光のコントラスト) を見つけ出す。
 - エリアの特徴を足し、隣接するエリアと競争する。

顕著性マップ例 2 – 3 (Saliency Map)



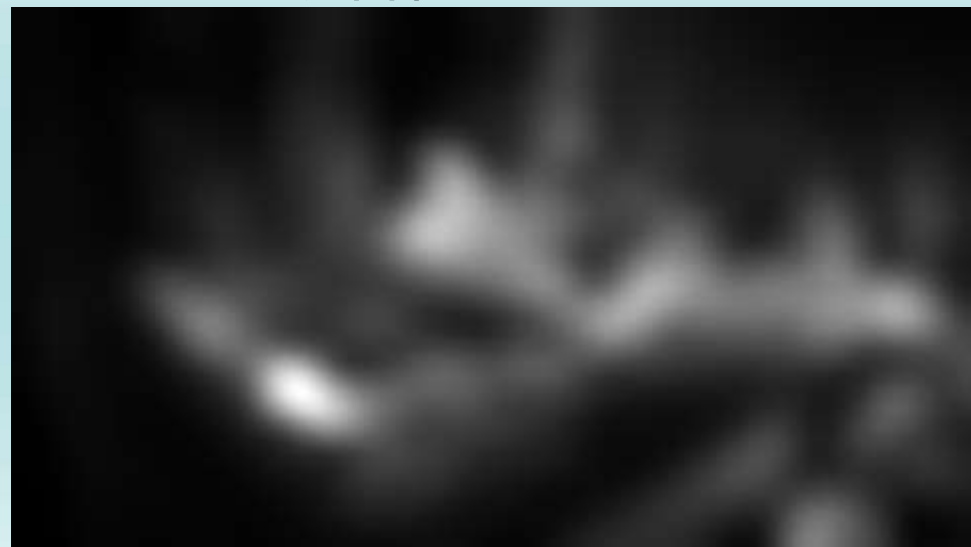
画像2



顕著性マップ2



画像3



顕著性マップ3

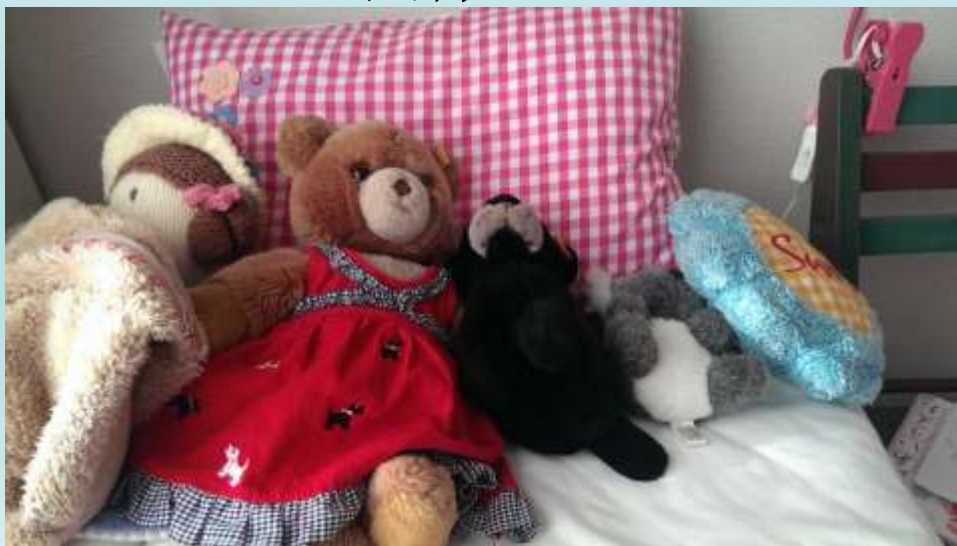
顕著性マップ例 4 – 5 (Saliency Map)



画像 4



顕著性マップ 4



画像 5



顕著性マップ 5

顕著性によりエラー隠蔽



目標: ネットワーク上で動画ストリーミングする時、
パケットが損失. 無くしたピクセルブロック \mathbf{b} を求め
る:

$$\min_{\mathbf{b}} fit_err(\mathbf{b})$$

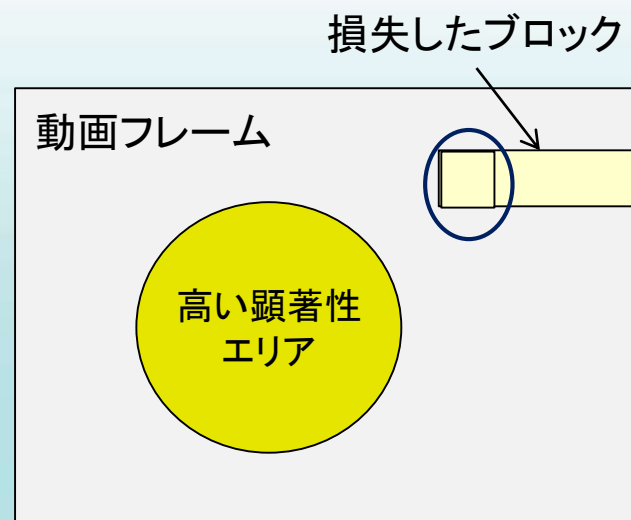
問題: 答えは複数である。

提案: 低い顕著性の成分を加える:

$$\min_{\mathbf{b}} \{fit_err(\mathbf{b}) + \lambda saliency(\mathbf{b})\}$$

長所:

1. 例え間違っても目立たない。
2. 損失したブロックの顕著性が低い可能性が高い。



顕著性によりエラー隠蔽

実験: 3.6dB improvement in PSNR.



従来技術RECAP



我々の提案

顕著性により画像修復 (image inpainting)



合成した仮
想視点画像

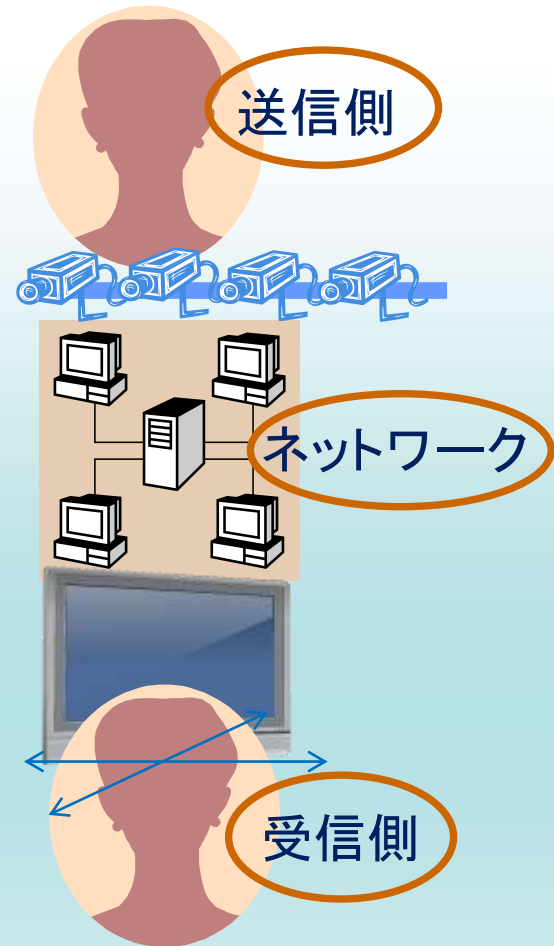
従来技術



修復した仮
想視点画像

我々の提案

まとめ



- ビジュアルコミュニケーションとは？
 - ネットワークを通して、どこの送信側からビジュアルメディアをどこの受信側までリアルタイムで送信。
 - 2次元動画から3次元動画まで、課題まだ残っている。
- スパース表現
 - 変換符号化
 - グラフ変換
 - 応用: 奥行き画像の圧縮、ノイズ除去、画像補間
- 顕著性マップ
 - 定義、例
 - 応用: エラー隠蔽、画像修復

Q&A

- 連絡先:
 - 電子メール宛先: cheung@nii.ac.jp
 - ホームページ: <http://research.nii.ac.jp/~cheung>