

省メモリ指向一枚超解像 アーキテクチャとそのFPGA実装

北海道大学大学院 情報科学研究科

大平 貴徳、真田 祐樹、築田 聡史、
五十嵐 正樹、池辺 将之、浅井 哲也、本村 真人

研究背景

- ・映像機器の高機能化に伴う映像の高解像化

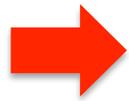
- レティナディスプレイ
- 4Kハイビジョンテレビ

- ・低解像度の映像コンテンツが数多く存在



解像度を高める研究(超解像)

- 高速
- 低コスト
- 解像度の精度



- ・省メモリ指向ハードウェアアーキテクチャ
- ・一枚超解像のFPGA実装

4Kテレビ



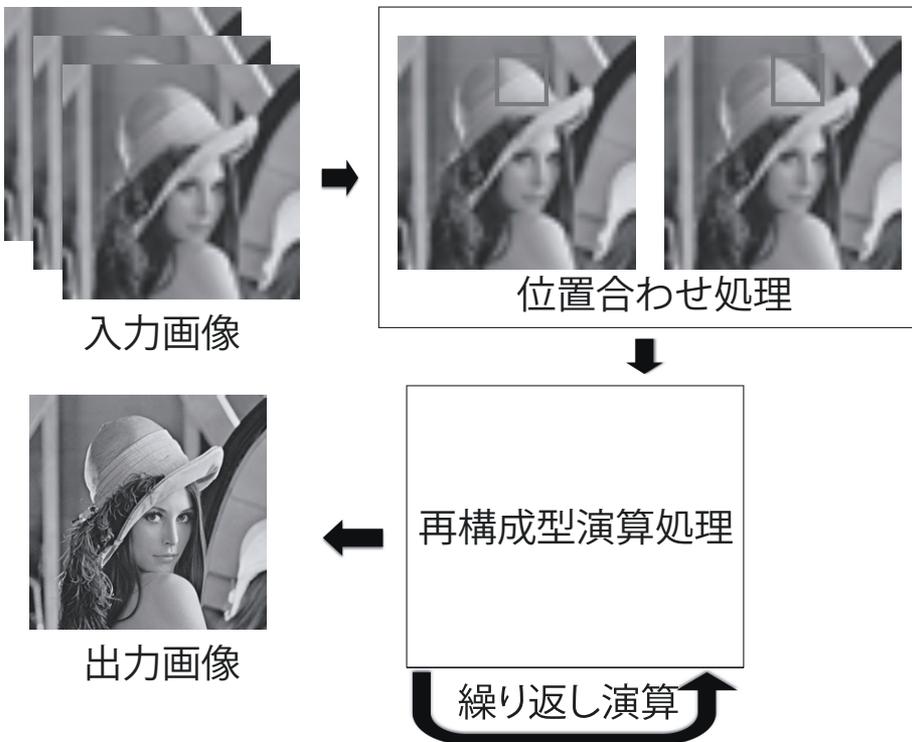
Super Resolution
image-processing



超解像の分類

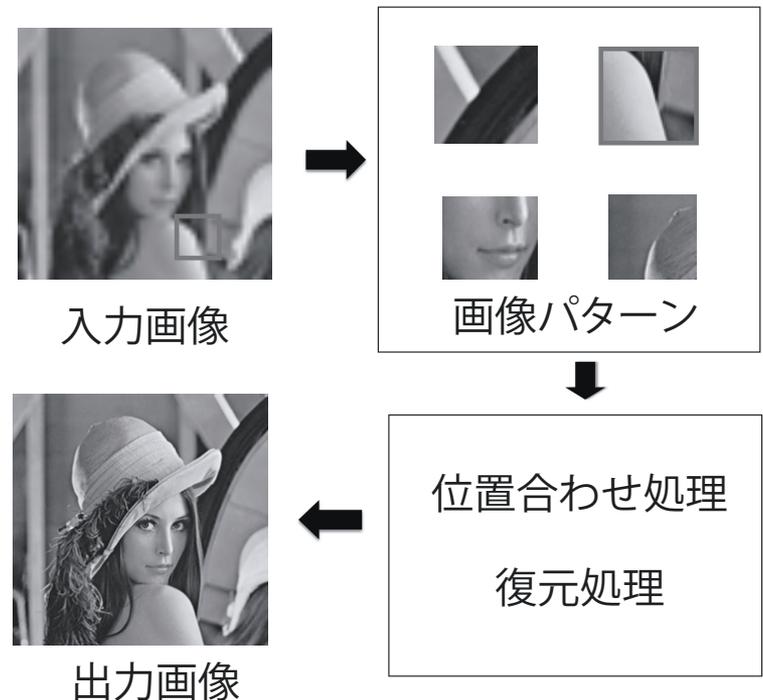
再構成型

- ① 映像や連写から複数枚の画像を用意
- ② それぞれの位置を合わせる
- ③ 画素を情報を対比させ繰り返しの処理を行う



学習型

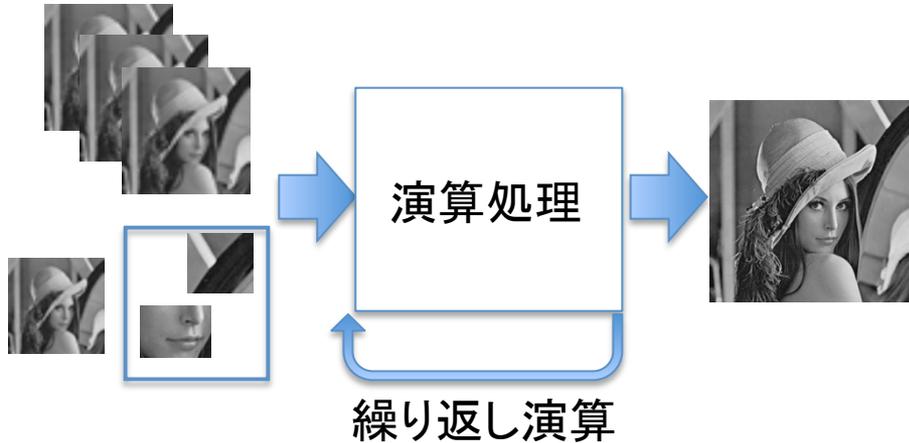
- ① 予め画像のパターンを学習
- ② 入力画像とパターンを比較
- ③ パターンにより画素情報を復元し出力



実装するアルゴリズムの位置づけ

従来の超解像

- ・複数枚の画像から高解像な画像を生成
- ・学習パターンから高解像な画像を生成

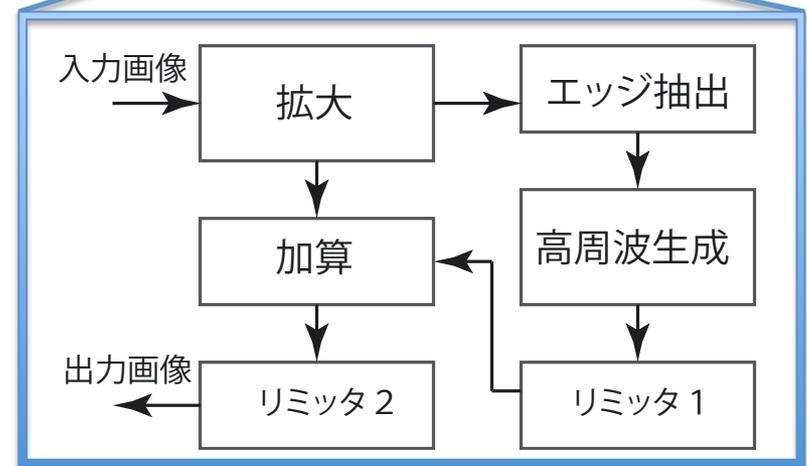
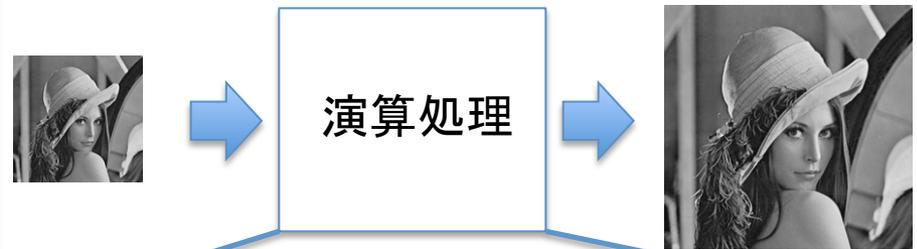


とても綺麗な出力が得られるが...

- 複数画像やデータベースを使用するので高コスト
- 位置合わせやイタレーションが必要なため演算量が多い

本研究で用いた一枚超解像

- ・一枚画像のみ使用
- ・繰り返しなしのストレートフォワード処理

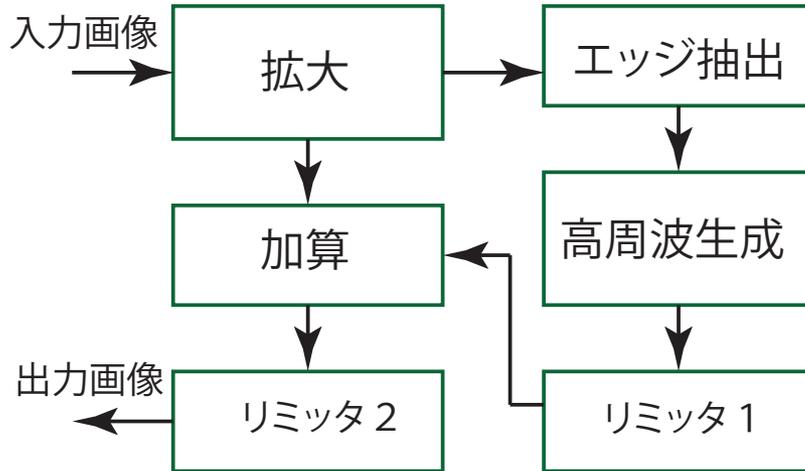


- ハードウェア指向
- 高速処理
- 低コスト

一枚超解像アルゴリズム

一枚超解像アルゴリズムフロー

- ① 縦横それぞれ2倍に拡大
- ② 拡大された画像にハイパスフィルタを施し、エッジ信号を抽出
- ③ 非線形処理によりエッジから高調波を取得
- ④ 発生した高調波を元画像に加算することにより高解像な出力を得る



拡大



[1]ではLanczos補間を用いて
画像の拡大を行っている



本研究ではこれを提案手法へ拡張

[1]S.Goshi, "A new signal processing method for video –Reproduce the frequency spectrum exceeding the Nyquist frequency-,” Proc. 3rd Multimedia Systems Conf.,pp.47-52,2012.

拡張した拡大部の説明

• Lanczos補間

sinc関数に基づく補間法

- Lanczos2フィルタ
 - 4行の入カラインバッファ
 - バイキュービック補間に類似
- Lanczos3フィルタ
 - 7行の入カラインバッファ
 - 理想的な補間



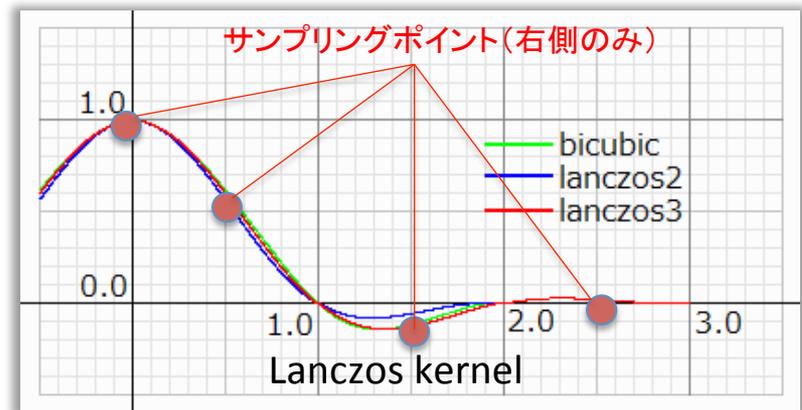
- Lanczos2ではジャギーが無視できない
- Lanczos3は高精度であるが計算コストや資源量が多い

• 考案手法

- 4行の入カラインバッファ
- ボックスフィルタリングに基づく処理
- 整数演算



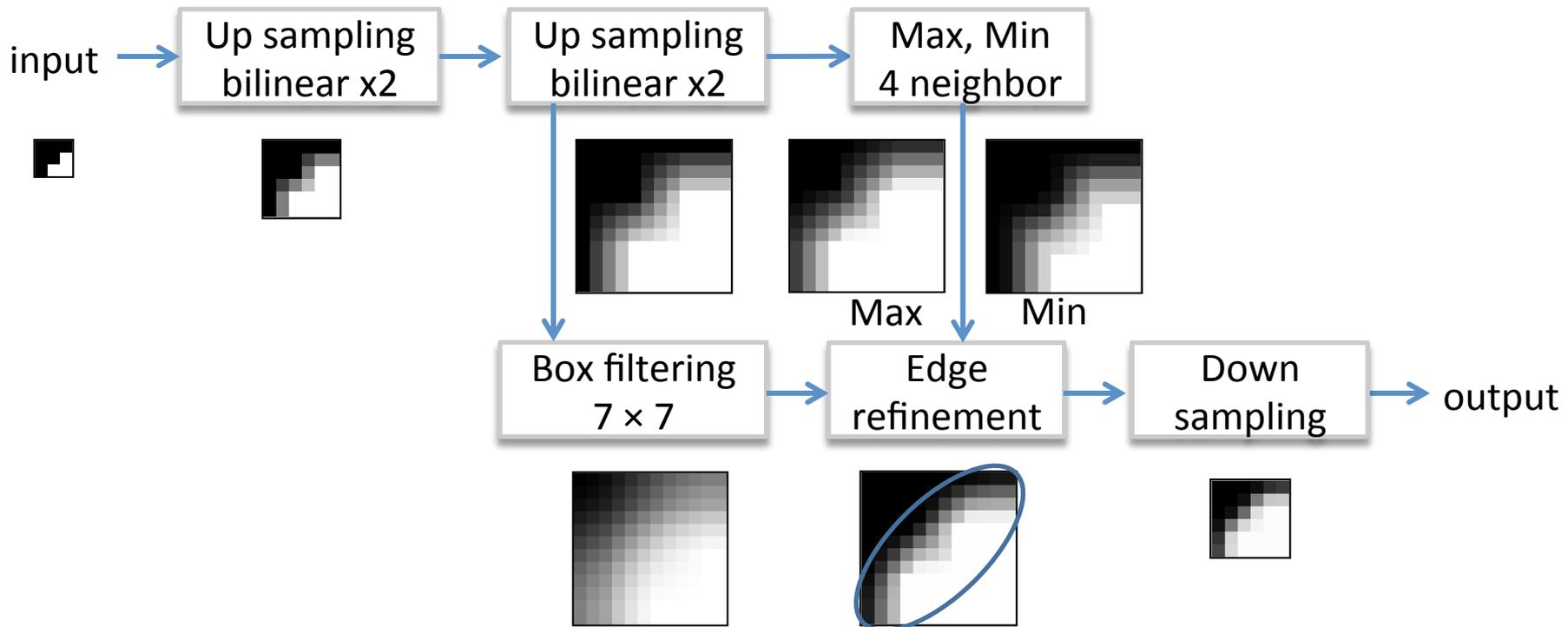
- ハードウェア指向
- 境界部におけるジャギーが少ない
- 低コストかつ高精度な補間が可能



B. Pribyl, "Fine image resampling algorithm"
Proceedings of CESC 2010

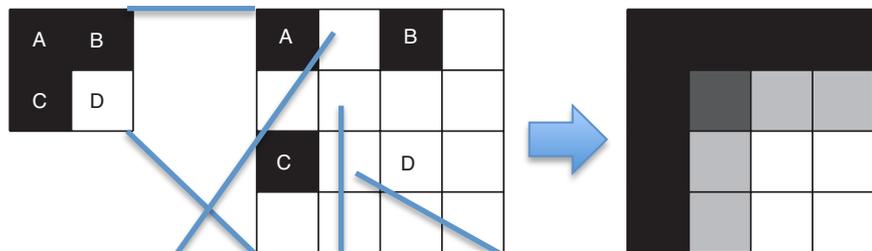
拡大アルゴリズムのフロー

- ① バイリニア補間を用いて入力画像を4倍する
- ② 拡大した画像の最隣4画素の最大値と最小値を算出
- ③ 拡大した画像に7×7の移動平均フィルタをかけ、ジャギーを除去
- ④ 算出した最大値、最小値を用いてジャギーを除去した画像のエッジを復元
- ⑤ 画像を半分に縮小し、最終的に入力画像の2倍の画像を出力



拡大: バイリニア拡大

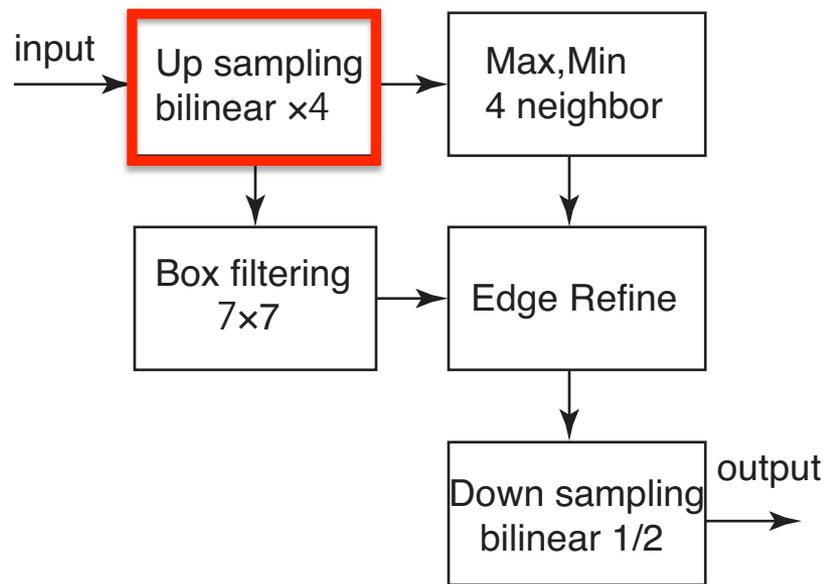
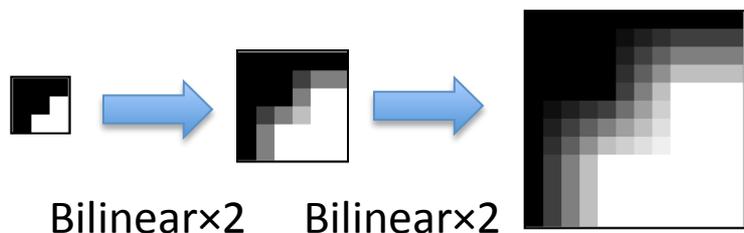
- ・線形補間を用いた拡大法
- ・再近傍の平均を取り画素を補間
- ・加算器、セレクタ、シフタにより構成



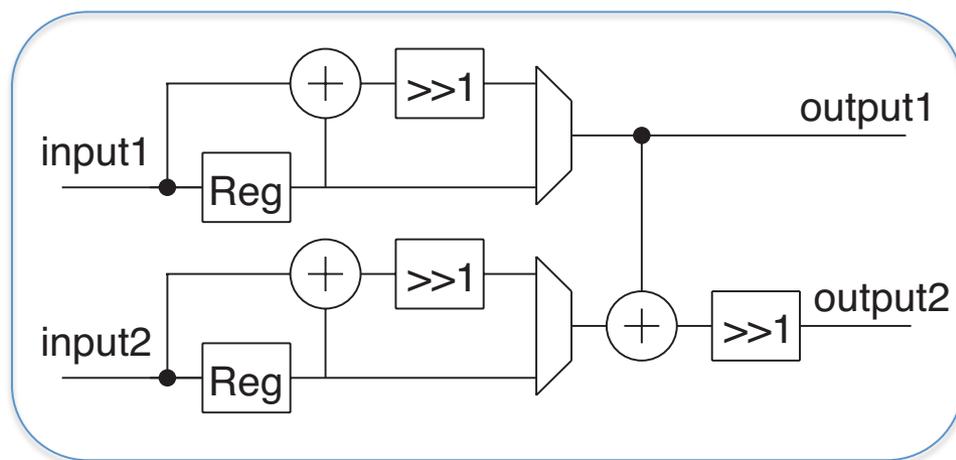
$$P_1 = \frac{A+C}{2},$$

$$P_3 = \frac{P_1 + P_2}{2},$$

$$P_2 = \frac{B+D}{2},$$

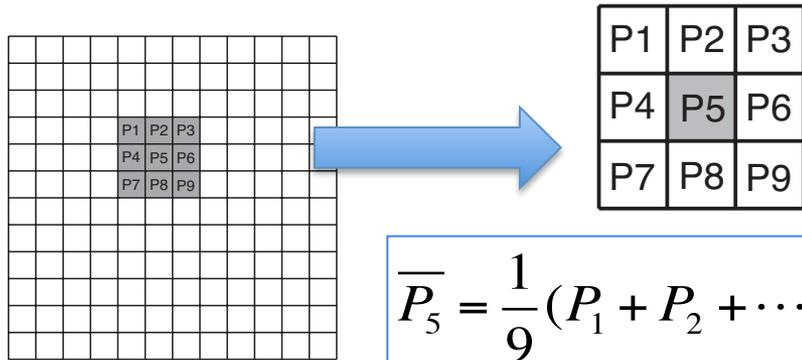


回路セル



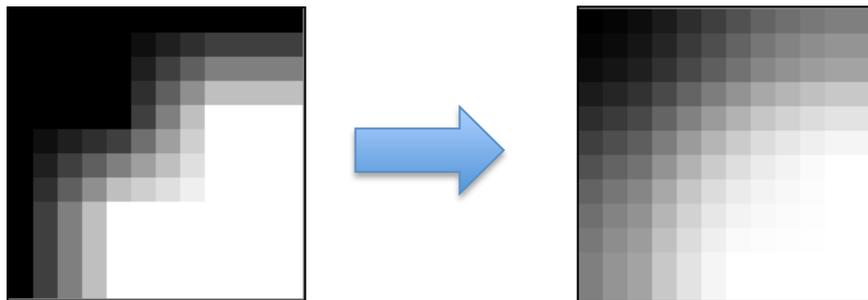
拡大: ボックスフィルタリング

- ・7×7の移動平均フィルタ
- ・ジャギーを除去しエッジを平滑化
- ・1行のラインバッファ、シフタ、加算器により構成

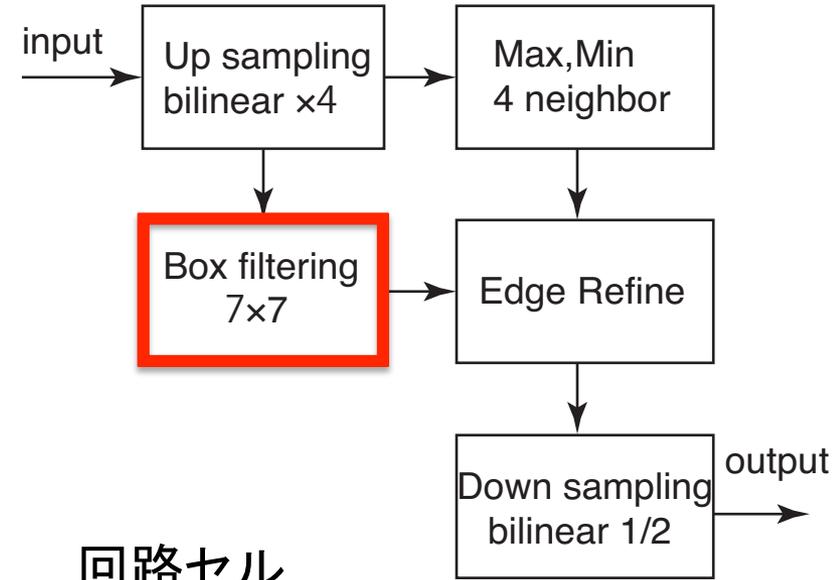


$$\bar{P}_5 = \frac{1}{9} (P_1 + P_2 + \dots + P_9)$$

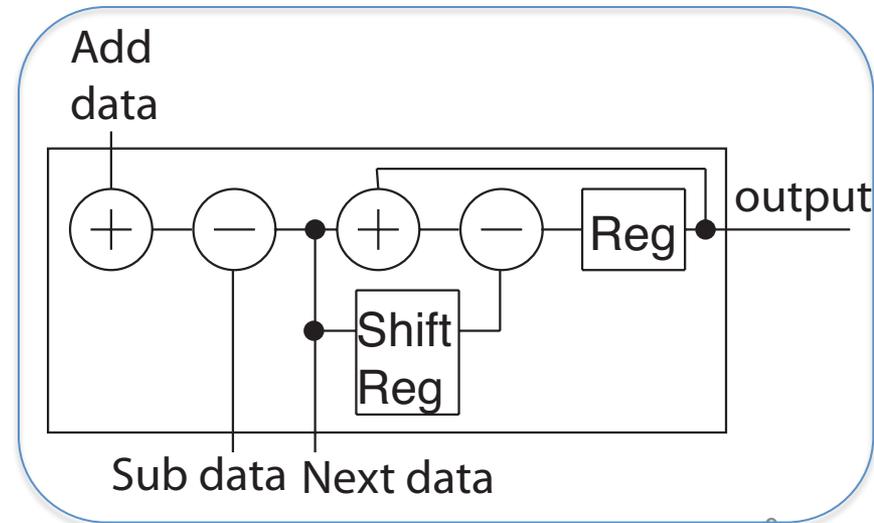
※3×3のカーネルの場合



boxfiltering 7×7

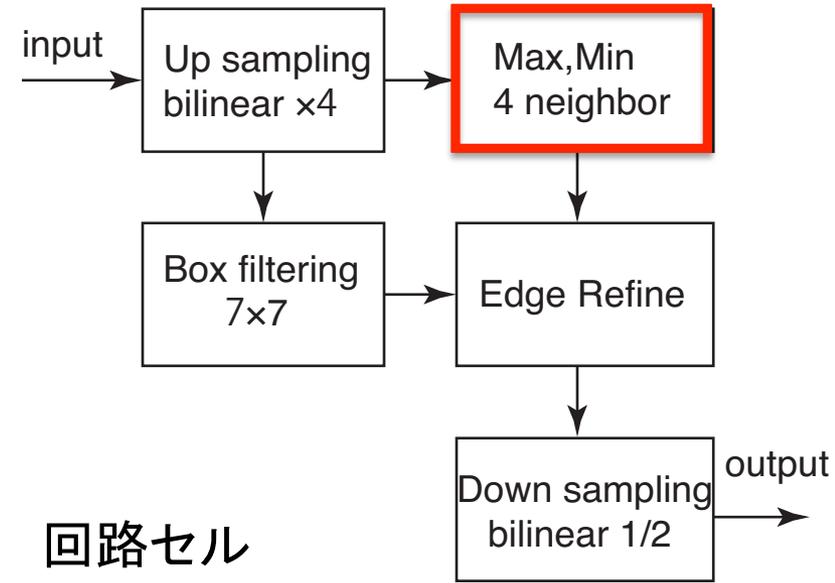
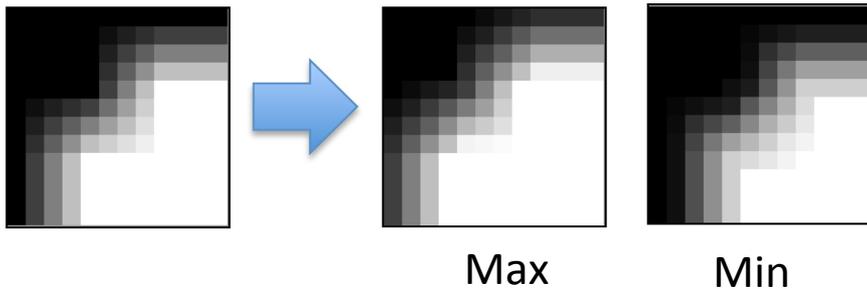
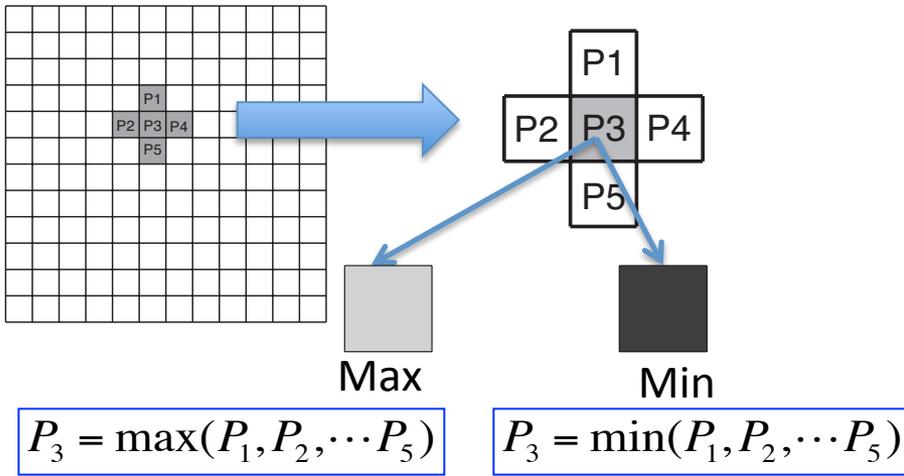


回路セル

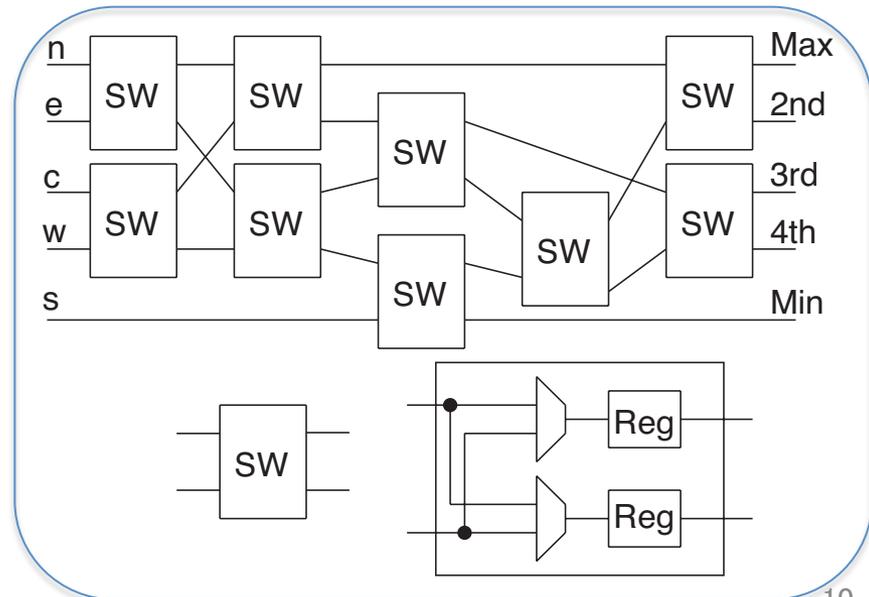


拡大: Max, Minの算出

- ・最近傍点における輝度の最大値と最小値を算出
- ・レジスタ、比較器により構成

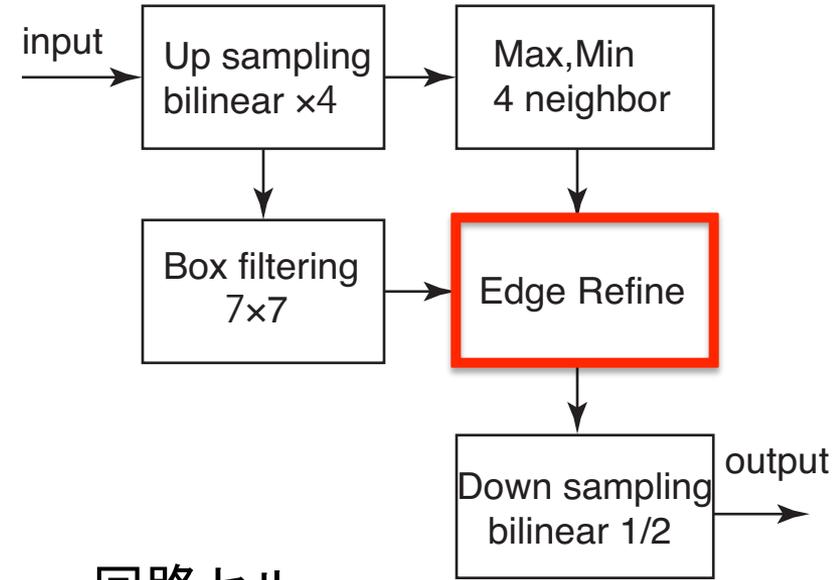


回路セル

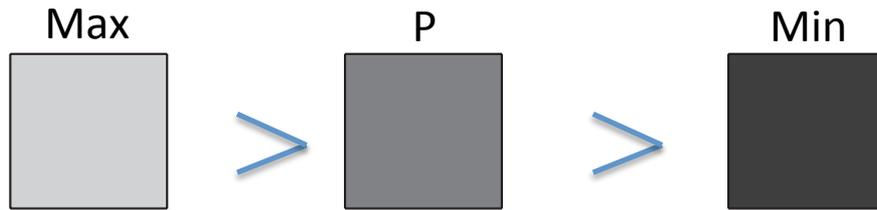
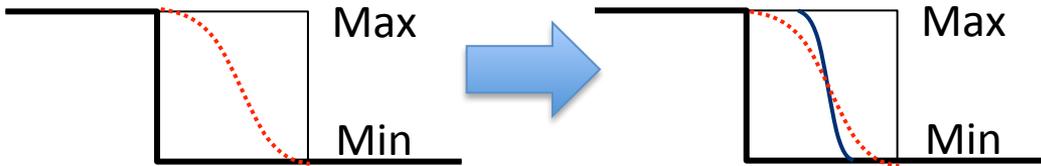
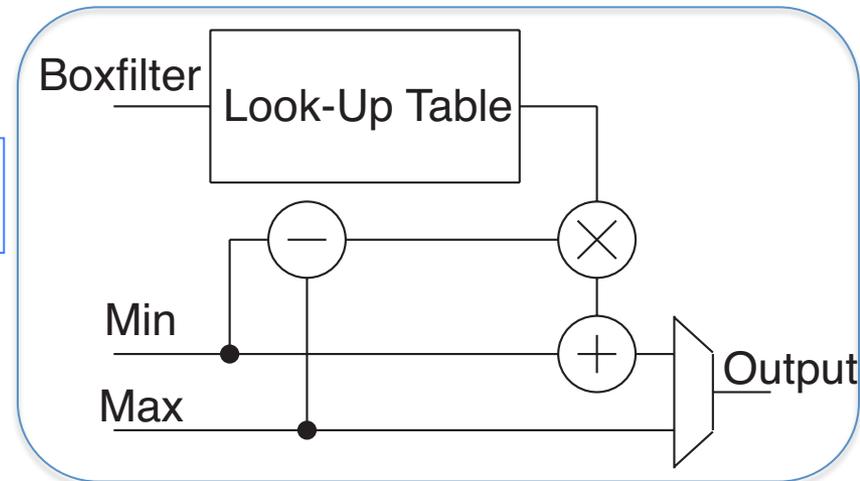


拡大: エッジリファイン

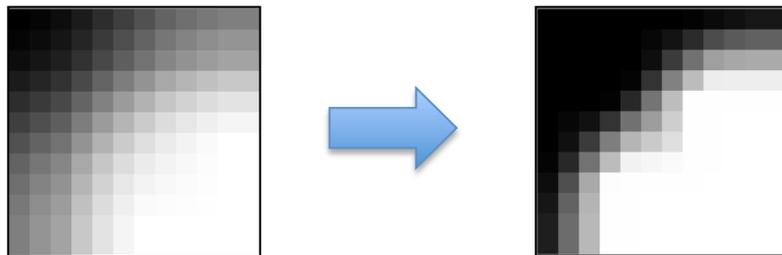
- ・最近傍における輝度の最大値と最小値の間で傾きを立ち上げる
- ・ルックアップテーブル、セレクタ、加算器により構成



回路セル



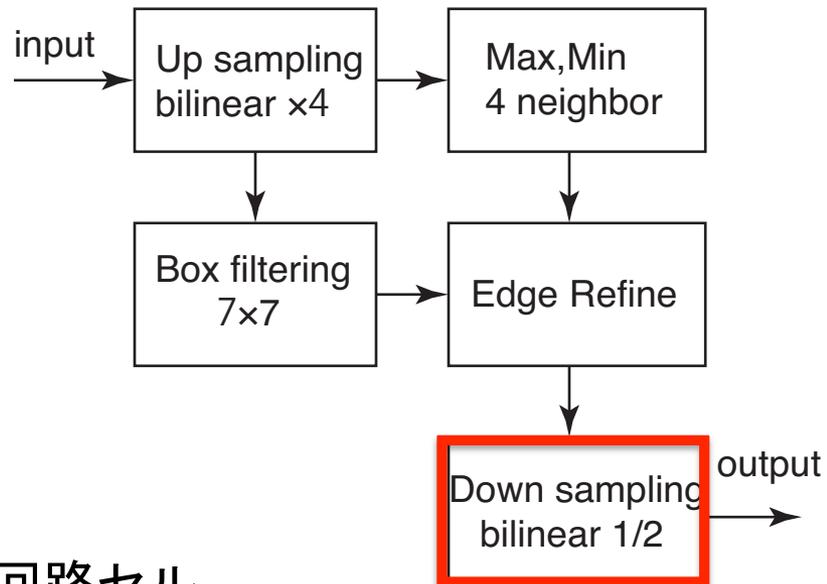
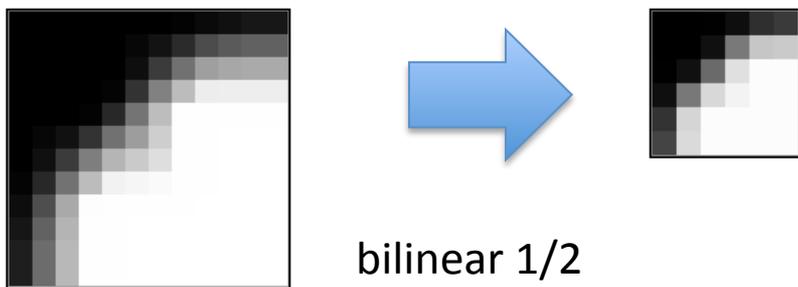
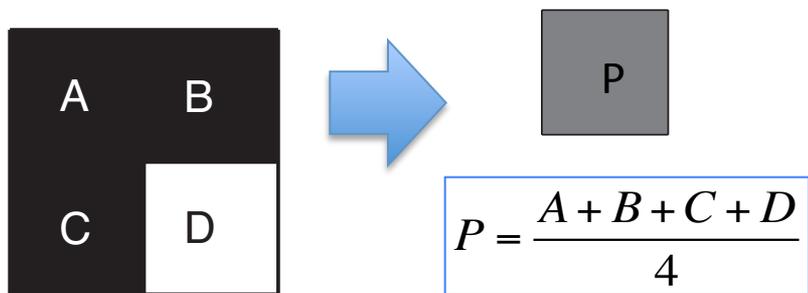
$$P' = \frac{(Max - Min) * (1 + \tanh(10 * (P / 255 - 0.5)))}{2} - Min$$



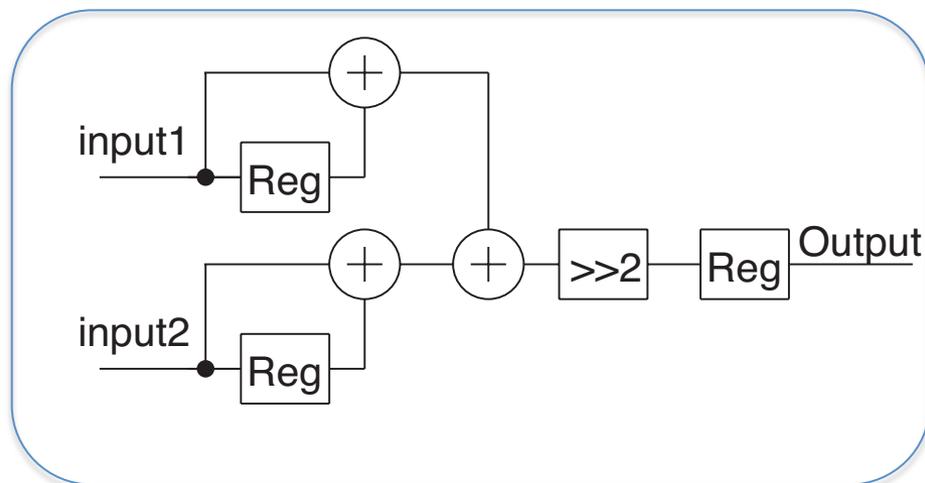
Edge Refine

拡大:バイリニア縮小

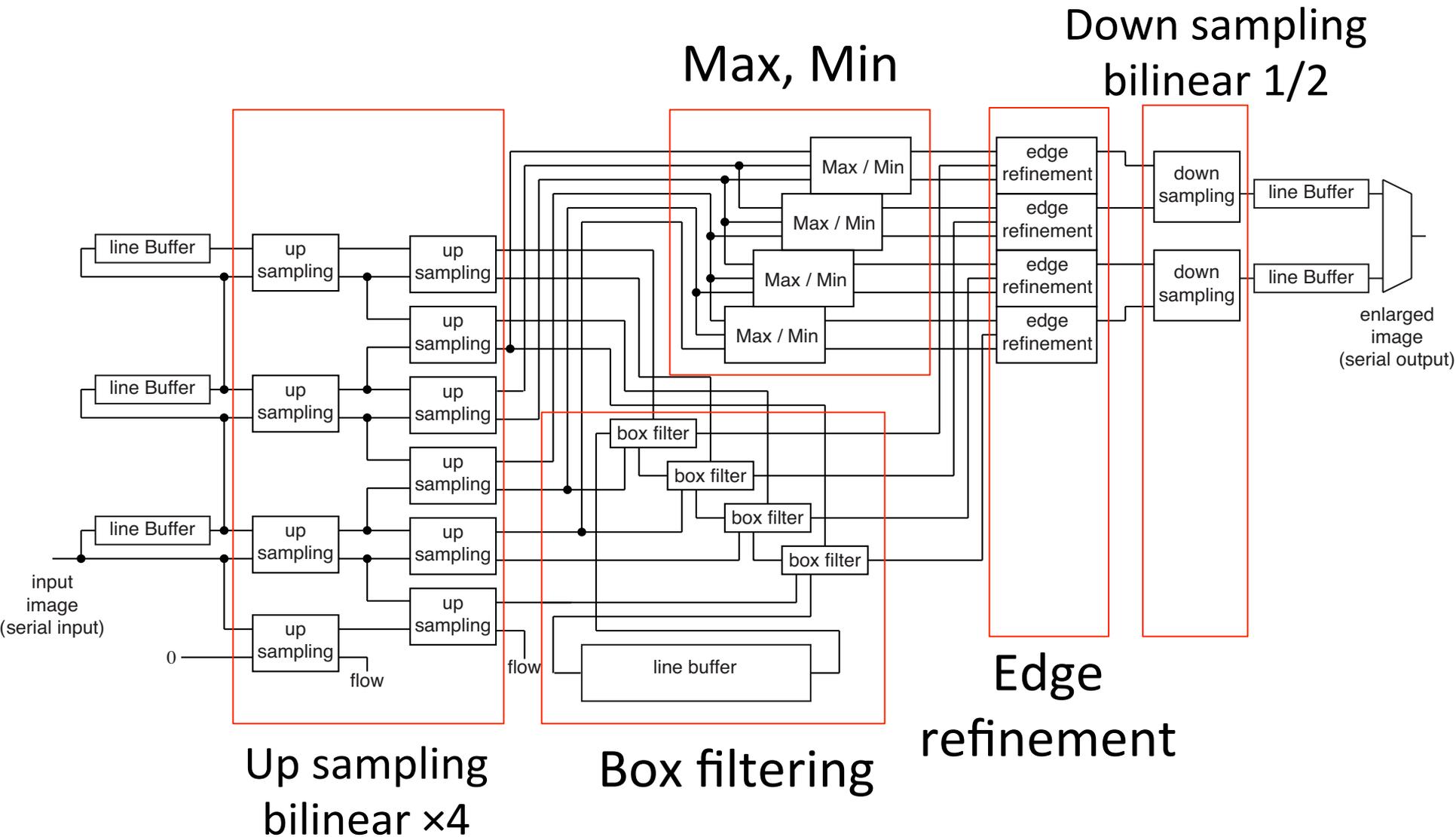
- ・線形補間を用いた画像の縮小
- ・近傍画素の平均を取り画像を縮小
- ・レジスタ、加算器、シフタにより構成



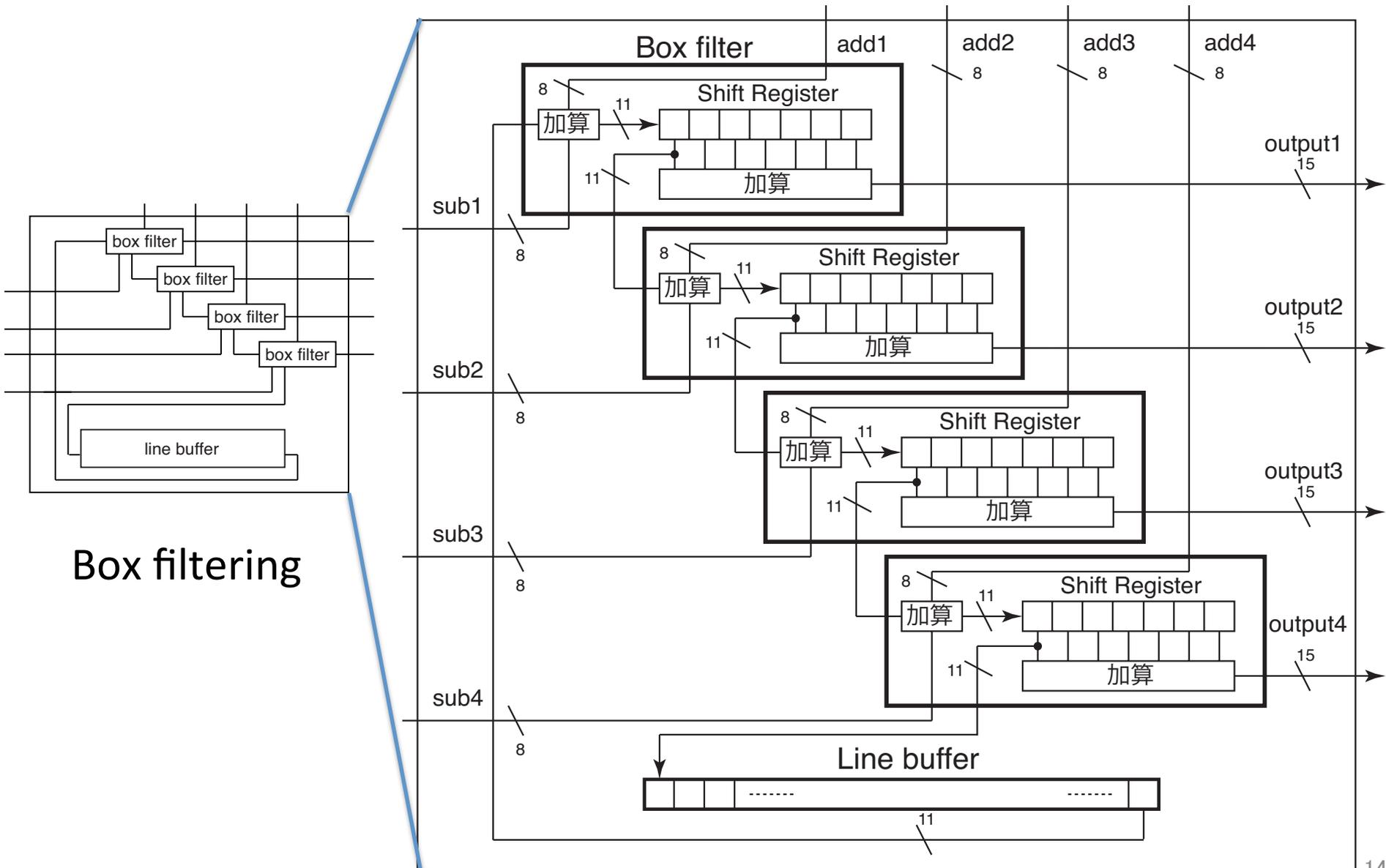
回路セル



拡大部回路構成図

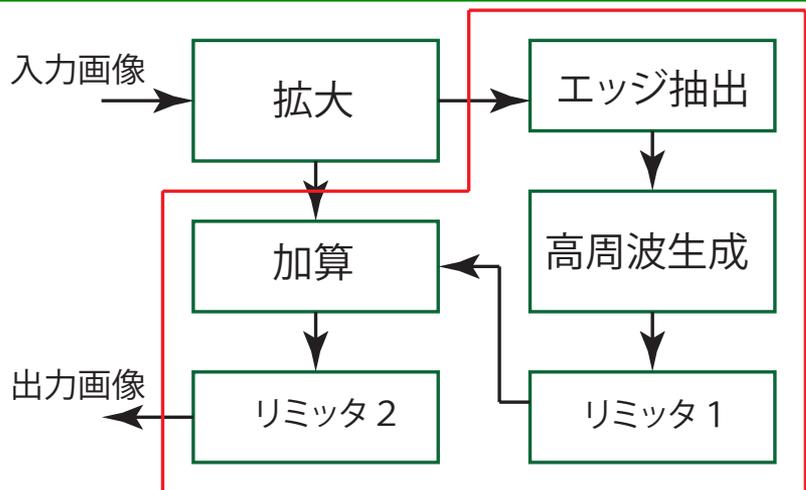


ボックスフィルタ回路構成図



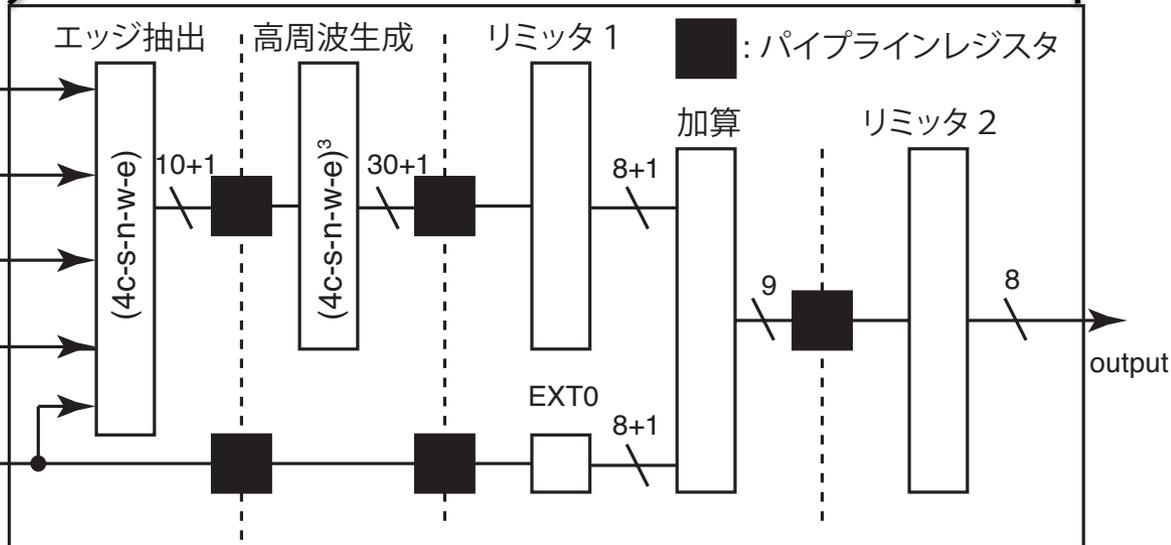
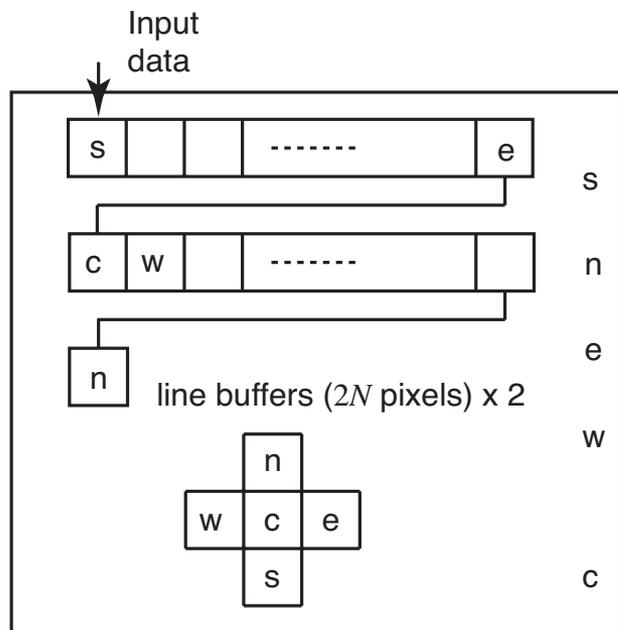
Box filtering

超解像回路構成図



$$c' = c + \alpha \left(c - \frac{4c + w + e + n + s}{8} \right)^3$$

$$= c + \frac{\alpha}{2^9} (4c - w - e - n - s)^3$$



FPGA実装結果

- FPGAについて

- ボード: PowerMedusa, MU300-DVI, StartiX II (Altera)
- ソフトウェア: Quartus II

- 拡大部

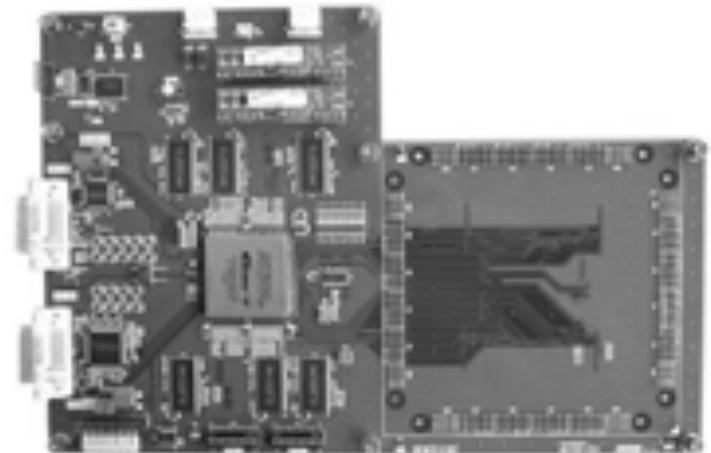
- 開発言語: Verilog HDL

Register	ALUT	Fmax
18,628	2,521	43.5MHz

- 超解像部

- 開発言語: VHDL、Verilog HDL
- DVI入出力
- 60fps, VGA

Register	ALUT	FPGA CLK
31,732	16,651	90MHz



FPGAボード MU300-DVI

画像出力結果

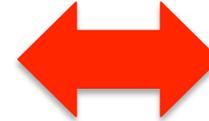
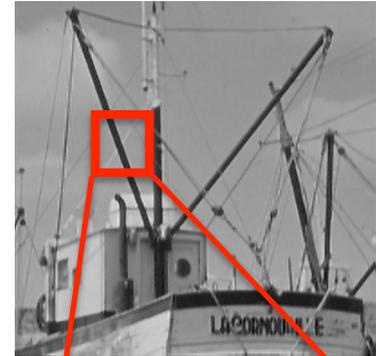
入力画像
(200 × 200)



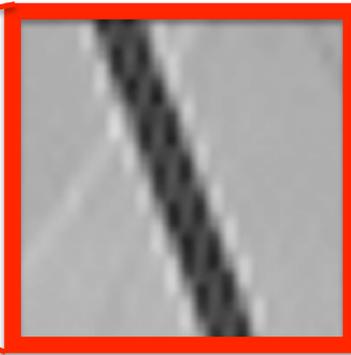
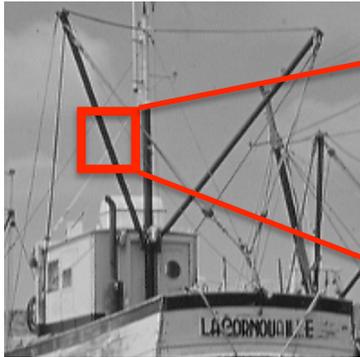
提案手法による拡大
(400 × 400)



超解像出力
(400 × 400)



バイキュービック補間+超解像出力
(400 × 400)



- ・8bitグレースケール
- ・200*200画素の画像
- ・提案手法とバイキュービック補間による超解像を比較



境界部におけるジャギーが大幅に
なくなり、高精度に高解像化

まとめ

研究の目的

- 高速かつハードウェア指向である一枚超解像のFPGA実装
- 画像の拡大における提案手法への拡張

FPGA実装結果

・拡大部

Register	ALUT	Fmax
18,628	2,521	43.5MHz

・超解像部

Register	ALUT	FPGA CLK
31,732	16,651	90MHz

画像出力結果

- 拡大部を提案手法に拡張した事で省メモリかつジャギーの少ない高精度な超解像を実現