

情報デザイン専攻

画像情報処理論及び演習I

**-画像合成-**  
*Etcetera*

第12回講義  
水曜日1限  
教室6218

吉澤 信  
shin@riken.jp, 非常勤講師  
大妻女子大学 社会情報学部

独立行政法人  
理化学研究所

Shin Yoshizawa: shin@riken.jp

レポートについて

**第3回**レポートの採点結果を返します!  
今 or 演習中に取りに来てください。

第1回と第2回の採点結果をまだ取りに来てない人も、取りに来てねー

- ✓ 採点に納得がいけない人は講義終了後に交渉可。
- ✓ レポートは7末(予定)まで受け付けますが、×切後は点数に0.8倍。
- ✓ **名前と学籍番号**かいてくださいねー(^.^);
- ✓ ファイル名・フォルダー名は半角英数でお願いします日本語(全角)はダメ。
- ✓ **提出後にzipやZIPの中身を必ず確認してください!**

Shin Yoshizawa: shin@riken.jp

今日の授業内容

[www.riken.jp/briect/Yoshizawa/Lectures/index.html](http://www.riken.jp/briect/Yoshizawa/Lectures/index.html)  
[www.riken.jp/briect/Yoshizawa/Lectures/Lec09.pdf](http://www.riken.jp/briect/Yoshizawa/Lectures/Lec09.pdf)

- ① 画像合成 Etcetera.
- ② 容量チェック.
- ③ 第4回レポートの説明.
- ④ 成績の計算について.
- ⑤ 授業評価アンケートについて.
- ⑥ 演習:前回に引き続きImage Analogy+Poisson Image Editing.
- ⑦ 最後10分:授業評価アンケート.

今日の演習は第4回レポートの内容なので  
みなさん頑張ってくださいねーp(^.^)q

Shin Yoshizawa: shin@riken.jp

Content-Aware Resizing/Retargeting

✓ 画像中の特徴的/象徴的オブジェクトのサイズやAspect Ratioを変えずに画像のサイズやAspect Ratioを変える事。

Shin Yoshizawa: shin@riken.jp

復習:勾配:Gradient

✓ 勾配(Gradient): スカラー場の各点で変化が最大の方向と変化率を大きさに持つベクトル場。

✓ 勾配作用素:  $\nabla = \left( \frac{\partial}{\partial x}, \frac{\partial}{\partial y} \right)$

✓ 勾配ベクトルの表記:  

$$\nabla I = \nabla I(x, y) = \left( \frac{\partial I}{\partial x}, \frac{\partial I}{\partial y} \right)$$

$$= \left( \frac{\partial I(x, y)}{\partial x}, \frac{\partial I(x, y)}{\partial y} \right) = (I_x, I_y)$$

✓ 勾配の大きさ:  

$$\|\nabla I\| = \sqrt{I_x^2 + I_y^2}$$

Shin Yoshizawa: shin@riken.jp

Content-Aware Resizing/Retargeting2

✓ Seam Carving: 勾配の強度エネルギーが最小になる曲線群を計算して複製・削除によりサイズ変更。

$$\nabla I = \left( \frac{\partial I}{\partial x}, \frac{\partial I}{\partial y} \right), \quad E(I) = \|\nabla I\|_L = \left| \frac{\partial I}{\partial x} \right| + \left| \frac{\partial I}{\partial y} \right|$$

Shin Yoshizawa: shin@nriken.jp

### Content-Aware Resizing3

✓ Seam Carving: ビデオへも拡張&マスクと組み合わせてオブジェクトの削除も。

Time

scams scale seams scale

©S. Arbel & A. Shamir, SIGGRAPH 2007

©M. Bahrami, SIGGRAPH 2008

Shin Yoshizawa: shin@nriken.jp

### Content-Aware Resizing/Retargeting4

✓ 勾配強度やマスクと他の画像特徴を組み合わせて局所変形(Warping)をする方法もある。

Gradient Map × Saliency Map = Significance Map

original image using gradient map using significance map

©Y. S. Wang et al., SIGGRAPH Asia 2008

©R. Gal et al., ECCV 10

Shin Yoshizawa: shin@nriken.jp

### 復習: 2-bands Blending

✓ 低周波は滑らかにAlphaを変化+高周波はAlpha定数。

低周波成分画像 高周波成分画像

©Blomer & Eusebi, 2003

Shin Yoshizawa: shin@nriken.jp

### イメージモザイク/パノラマ画像生成

- 特徴点・対応点抽出。
- 幾何学的変換の推定。
- 幾何変換&色補正。
- Blending.

高周波 低周波

特徴中心 投影中心

投影面上での貼り合わせ

上から見た図

©CC-AUTV研究会

Shin Yoshizawa: shin@nriken.jp

### 復習: Pixel TransferによるInpainting

✓ 画像から似ている画素・Textureを持つてくる。

- 局所Windowで類似パターンを検索: Windowサイズに依存。
- 低周波画像は補間で生成しておくこと影等の効果を反映出来る。
- 穴(マスク)を埋める順番が重要!

類似検索

©D. Hoehn, Univ. Illinois

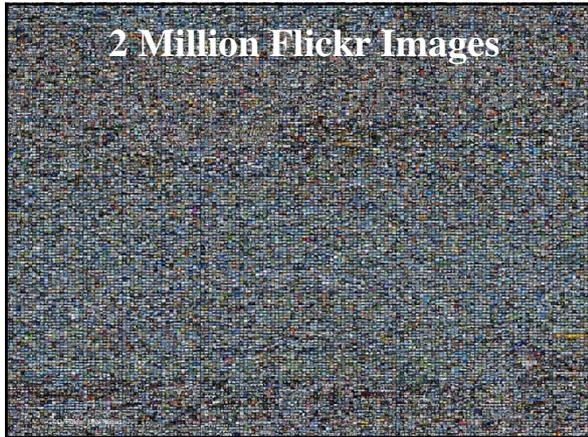
©R. Yamachi et al., CED 2003

Shin Yoshizawa: shin@nriken.jp

### 数百万のExampleがあると...

✓ データベースからの類似検索+画像合成。

©D. Hoehn, Univ. Illinois



Shin Yoshizawa: shin@riken.jp

## 数百万のExampleがあると…

©D. Hahn, Univ. Illinois

Shin Yoshizawa: shin@riken.jp

## 数百万のExampleがあると…

©D. Hahn, Univ. Illinois

Shin Yoshizawa: shin@riken.jp

## Image2GPS

✓ 画像データベースに位置情報が付いている(撮影場所)と画像検索→類似画像の多さを地図上にマップ=GPS.

©Mays & Elins, C/PPH 2008

Shin Yoshizawa: shin@riken.jp

## Image2Real

✓ 画像データベースからCGの類似画像検索+部分合成により写実的(Photorealistic)なCGの生成.

Input → Similar images → Chroma keying → Local style transfer → Output

Image Database, Similar images, Chroma keying, Local style transfer, Output

©MT-CSAIL-TR 2009-034

Shin Yoshizawa: shin@riken.jp

## カモフラージュ画像合成

Camouflage Art:

©New York Times, ©H.K. Cho, PHOTOGRAPHY 2010, ©K. Gardner, ©J. Sirosh

Shin Yoshizawa: shin@riken.jp

## 復習: Image AnalogyのTexture Synthesis

Shin Yoshizawa: shin@riken.jp

## カモフラージュ画像合成

©H.-K. Chu, SIGGRAPH 2010

Shin Yoshizawa: shin@riken.jp

## 今日の授業内容

[www.riken.jp/brict/Yoshizawa/Lectures/index.html](http://www.riken.jp/brict/Yoshizawa/Lectures/index.html)  
[www.riken.jp/brict/Yoshizawa/Lectures/Lec09.pdf](http://www.riken.jp/brict/Yoshizawa/Lectures/Lec09.pdf)

- ① 画像合成 Etcetera.
- ② 容量チェック.
- ③ 第4回レポートの説明.
- ④ 成績の計算について.
- ⑤ 授業評価アンケートについて.
- ⑥ 演習: 前回に引き続きImage Analogy+Poisson Image Editing.
- ⑦ 最後10分: 授業評価アンケート.

今日の演習は第4回レポートの内容なので  
みなさん頑張ってくださいねーp(^^)q

Shin Yoshizawa: shin@riken.jp

## 容量チェック

- ✓ みなさんのLinuxでのディスク容量の上限は **1GB=1000MB**です.
- ✓ 上限を超えてのデータ作成は出来ない=レポート出来ない  
ので、余分な画像・レポートデータは圧縮するか、USB  
メモリーキー等にバックアップを取って削除しましょう！
- ✓ 自分がどのくらいのハードディスクのデータ容量を  
使っているかを確認する方法: 端末で「`du -h ~`」と打  
ち込んでエンターキーを押す.
- ✓ フォルダ・ディレクトリーの圧縮方法: ファイルブラウ  
ザーで右クリックして圧縮を選択. または、端末で圧縮  
したいフォルダの一つ上のフォルダまで「`cd 圧縮した  
いフォルダの一つ上のフォルダ`」で移動した後、端末  
で「`zip -r 圧縮したいフォルダ名 zip 圧縮したいフォル  
ダ名`」で圧縮出来ます.

↑は授業のHPにもあります

[www.riken.jp/brict/Yoshizawa/Lectures/index.html](http://www.riken.jp/brict/Yoshizawa/Lectures/index.html)

Shin Yoshizawa: shin@riken.jp

## 第4回レポート

[www.riken.jp/brict/Yoshizawa/Lectures/index.html](http://www.riken.jp/brict/Yoshizawa/Lectures/index.html)

- ✓ 内容は画像合成でImage AnalogyとPoisson Image Editing  
の演習内容です.
- ✓ **第4回レポートの〆切は7月20日**です.

授業のHPで公開

Shin Yoshizawa: shin@riken.jp

## レポートについて

**第3回**レポートの採点結果を返します！  
演習中に取りに来てください.

[第1回と第2回の採点結果をまだ取りに来てない人も、取りに来てねー](#)

- ✓ 採点に納得がいけない人は講義終了後に交渉可.
- ✓ レポートは7末(予定)まで受け付けますが、〆切後は点数に  
0.8倍.
- ✓ **名前と学籍番号**かいてくださいねー(^^);
- ✓ ファイル名・フォルダ名は半角英数でお願いします日本語  
(全角)はダメ.
- ✓ **提出後にzipやZIPの中身を必ず確認してください!**

Shin Yoshizawa: shin@riken.jp

## 成績の計算方法について

出席4割、レポート6割.

成績点数(基本は...もしかしたらゲタが...)=  
 出席日数 × (20/7) + レポート4回分の合計点 × (3/20)

- ✓ 出席点は40点を休講を抜かした14で割って一回あたり(20/7)点です。補講日も加算します。遅刻した日は(20/7)の代わりに $0.8 \times (20/7)$ で計算してください。
- ✓ レポートの点は60点を400点で割ってレポートの1点あたり(3/20)点です。
- ✓ 合計点の小数点以下は切り上げします。合計点100点以上の人は100点です(^.^);
- ✓ S: 100-90点, A: 89-80点, B: 79-70点, C: 69-60点, D: 59-0点。
- ✓ 単位取得ボーダーの人や「あと数点で一つ上の評価なので何とか...」という人は補講日(7/29:5限)に相談可。

↑は授業のHPIにもあります



Shin Yoshizawa: shin@riken.jp

## 授業評価アンケートについて

- ✓ 授業評価アンケートを取りに来てください!
- ✓ 今日最後の10分は授業評価アンケートを書いてください。

回収してくれる1、2人募集!  $m(\underline{\quad})_m$

- ✓ ↑だれかお願いします  $p(\geq \square \leq)q$ .
- ✓ 3階の助手さん部屋に提出ボックスがあるので授業後に集めて持って行ってほしいです。

Shin Yoshizawa: shin@riken.jp

## 演習

[www.riken.jp/briect/Yoshizawa/Lectures/index.html](http://www.riken.jp/briect/Yoshizawa/Lectures/index.html)  
[www.riken.jp/briect/Yoshizawa/Lectures/Lec09.pdf](http://www.riken.jp/briect/Yoshizawa/Lectures/Lec09.pdf)

### Image Analogy+Poisson Image Editing.

Image Analogy:  
[www.riken.jp/briect/Yoshizawa/Lectures/Lec07.pdf](http://www.riken.jp/briect/Yoshizawa/Lectures/Lec07.pdf)  
[www.riken.jp/briect/Yoshizawa/Lectures/Ex05.zip](http://www.riken.jp/briect/Yoshizawa/Lectures/Ex05.zip)

Poisson Image Editing+(NumberEditor):  
[www.riken.jp/briect/Yoshizawa/Lectures/Lec08.pdf](http://www.riken.jp/briect/Yoshizawa/Lectures/Lec08.pdf)  
[www.riken.jp/briect/Yoshizawa/Lectures/Ex06.zip](http://www.riken.jp/briect/Yoshizawa/Lectures/Ex06.zip)

今日の演習は第4回レポートの内容なので  
 みなさん頑張ってくださいねーp(^.^)q

Shin Yoshizawa: shin@riken.jp

## 演習:pgmとppmの変換

- ✓ jpgやbmp画像フォーマットから演習で使っている非圧縮のpgmやppmに変換するには...端末で

```
convert -quality 100 -compress none -depth 8 -comment "" INPUT.jpg OUTPUT.ppm
```

と打ち込んでエンターキーを押す。INPUT.jpgが変換したいファイル名でOUTPUT.ppmが(ppmに)変換後のファイル名。

- ✓ レポートでWordで使うためには、逆の変換をする、端末で

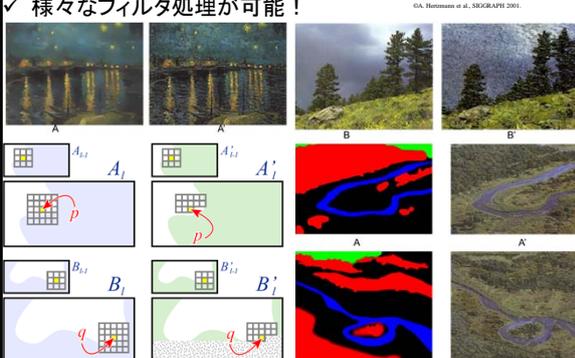
```
convert -quality 100 INPUT.ppm OUTPUT.bmp
```

と打ち込んでエンターキーを押す。INPUT.ppmが変換したいファイル名でOUTPUT.bmpが(bmpに)変換後のファイル名。

Shin Yoshizawa: shin@riken.jp

## 復習:Image Analogy

✓ 様々なフィルタ処理が可能!



Shin Yoshizawa: shin@riken.jp

## 重要:演習:MaskEditor & NumberEditor

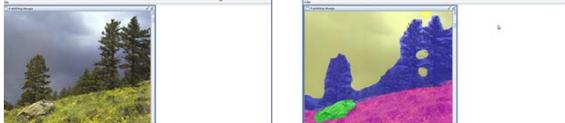
1. 端末にて「tcsh」と打ち込んでエンターキー。
2. 端末にて「setenv LANG C」と打ち込んでエンターキー。
3. 「sh Run\_MaskEditor.sh」 or 「sh Run\_NumberEditor.sh」

Shin Yoshizawa: shin@nken.jp

### 演習: NumberEditor & TextureByNumbers

- Image Analogy用TextureByNumbersのお絵かきGUI (Java).
- Ex06/NumberEditor/

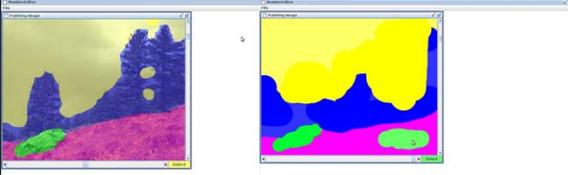
- sh Run\_NumberEditor.shで立ち上げてください.
- 画像を読み込む: File->Load ppm Image. Ex05/darkclouds.ppmを開いてみてください.
- お絵かき: 左ドラッグ: 木、岩、草原、空を違う色で塗ってみてください.
  - 色を変える: 右下のSelectボタン.
  - ブラシのサイズを変える: 右のスクロールバー or マウスホイール.
  - 表示の透明度を変える: 下のスクロールバー.
- セーブ:File->Save Number Image. A.ppmという名前で保存してください.



Shin Yoshizawa: shin@nken.jp

### 演習: NumberEditor & TextureByNumbers

- 追加のお絵かき: 木、岩、草原、空で使った色とほぼ同じ色で書き足してみてください.
- マスク画像(ppm)をセーブ:File->Save Number Image. B.ppmという名前で保存してください.
- A.ppmとB.ppmをEx05の下に移動(コピーでもカット&ペーストでもOK)してください.
- 端末を新たに立ち上げて、Ex05にcdで移動してください. もしもEx05をデスクトップで立ち上げていたら「cd ~/Desktop/Ex05」又はファイルブラウザのパスをコピーして端末に張り付けて「cd パス」でエンターキーを押す.



Shin Yoshizawa: shin@nken.jp

### 演習: NumberEditor & TextureByNumbers

- emacsでRun\_TextureByNumbers.shを立ち上げて、以下の様に書き換えてください. /TextureByNumbersの後の
  - 第一引数oxbow-mask.ppm は A.ppm
  - 第二引数oxbow.ppm は darkclouds.ppm
  - 第三引数oxbow-newmask.ppm は B.ppm
  - その後のppmファイル名も上のルールで変更してください.
- Run\_TextureByNumbers.shをセーブ(上書き保存)してください.
- 端末にて「sh Run\_TextureByNumbers.sh」で実行してみてください.

**訂正あり**



こんな感じなのが  
出れば正解⇒

Shin Yoshizawa: shin@nken.jp

### 演習: Image Analogy: Artistic Filters

- 端末でEx05に移動:もしもEx05をデスクトップで立ち上げていたら「cd ~/Desktop/Ex05」又はファイルブラウザのパスをコピーして端末に張り付けて「cd パス」でエンターキーを押す.

- 油絵フィルタをlena.ppmに適用してみる: 端末で、
 

```
./ArtisticFilter rhone-src.ppm rhone.ppm lena.ppm test1.ppm 1.0 1000.0 2
```

 を打ち込んでエンターキーを押す. 実行が終了したら、 端末で、
 

```
display test1.ppm &
```

 同様、
  - 水彩画フィルタをlena.ppmに適用してみる:
 

```
./ArtisticFilter watercolor-src.ppm watercolor.ppm lena.ppm test2.ppm 1.0 1000.0 2
```
  - 線画フィルタをlena.ppmに適用してみる:
 

```
./ArtisticFilter squire-blur.ppm squire.ppm lena.ppm test3.ppm 1.0 1000.0 2
```

**訂正あり**



A	A'	B	B'

注意: この例は全てTexture度2.0での実行結果.

**訂正あり**

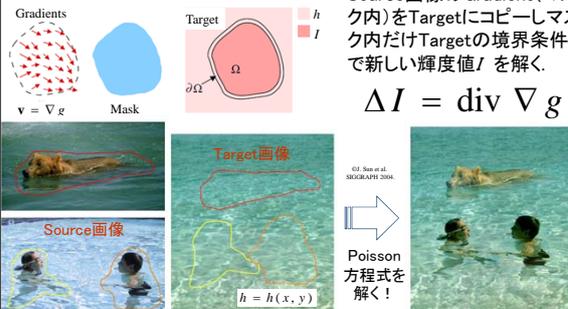


Shin Yoshizawa: shin@nken.jp

### 復習: Poisson Image Editingの原理

- Idea: 良いBlendingはSource画像の勾配(Gradient=エッジ)を可能な限り保持する事が重要.

Source画像のGradient(マスク内)をTargetにコピーしマスク内だけTargetの境界条件で新しい輝度値Iを解く.

$$\Delta I = \text{div } \nabla g$$


Source画像:  $v = \nabla g$

Mask

Target

Target画像

Source画像

$h = h(x, y)$

Poisson方程式を解く!

Shin Yoshizawa: shin@nken.jp

### 演習:MaskEditor

✓ PIE用マスク作成GUI (Java): Ex06/MaskEditor/

- sh Run\_MaskEditor.shでMaskEditorを立ち上げてください。
- Source画像を読み込む: File->Load SourceでEx06/images/Keira02.ppmを開いてください。
- Target画像を読み込む:File->Load TargetでEx06/images/MonaLisa.ppmを開いてください。
- 左クリックでPolylineを生成してKeiraの顔領域を作成してみましょう!

Shin Yoshizawa: shin@nken.jp

### 演習:MaskEditor

✓ PIE用マスク作成GUI (Java): Ex06/MaskEditor/

1. Source画像の大きさと位置を合わせる: Keiraの顔とMonaLisaの顔の大きさと位置を合わせてみよう!

- 右クリックでMove Picを選べば平行移動可能。
- 右クリックでAddを選べばPolyline作成モードに戻る。
- マウスの真ん中ホールで拡大縮小。
- Polylineの頂点は左クリックで移動可能。
- 下のスクロールバーで表示の透明度を変更可能。

Shin Yoshizawa: shin@nken.jp

### 演習:MaskEditor

- マスク画像(pgm)とTargetと同じ大きさのSource画像(ppm)の二つの画像をセーブ: File->Save Masks: ソースとマスクをKeiraMonaという名前でセーブしてみよう!

注:セーブするファイル名に拡張子はいらぬ:ファイル名.pgmとファイル

- 端末でPoissonImageEditorを以下の様に動かして合成してみよう!
  - 端末を立ち上げてEx06へ移動:「cd ~/Desktop/Ex06」.
  - ./PoissonImageEditor ./MaskEditor/KeiraMona.ppm ./MaskEditor/KeiraMona.pgm ./images/MonaLisa.ppm KM\_PIE.ppm 1.0 0.0
  - display KE\_PIE.ppm &

Source    Mask    Target    合成結果

Shin Yoshizawa: shin@nken.jp

### 演習:Image Analogy+PIEでリアルな合成

EdgePreservingFilterとArtisticFilterを使ってよりリアルな合成をしてみよう!

- EdgePreservingFilterでMonaLisa.ppmをフィルタリング: 端末で  
~/Desktop/Ex05/EdgePreservingFilter ./images/MonaLisa.ppm ML\_EP.ppm 1.0
- display ML\_EP.ppm &

Shin Yoshizawa: shin@nken.jp

### 演習:Image Analogy+PIEでリアルな合成

EdgePreservingFilterとArtisticFilterを使ってよりリアルな合成をしてみよう!

- MonaLisa.ppmの代わりにML\_EP.ppmで合成(ソース、マスクはそのまま):  
./PoissonImageEditor ./MaskEditor/KeiraMona.ppm ./MaskEditor/KeiraMona.ppm ML\_EP.ppm KM\_PIE\_EP.ppm 1.0 0.0
- display KM\_PIE\_EP.ppm &

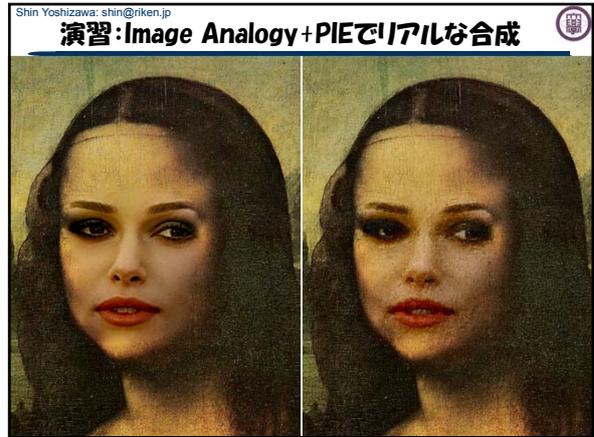
Source    Mask    Target    合成結果

Shin Yoshizawa: shin@nken.jp

### 演習:Image Analogy+PIEでリアルな合成

- Artistic Filterで細部を復元: A: ML\_EP.ppm A':./images/MonaLisa.ppm B:KM\_PIE\_EP.ppm  
~/Desktop/Ex05/ArtisticFilter ML\_EP.ppm ./images/MonaLisa.ppm KM\_PIE\_EP.ppm result.ppm 2.0 1000.0 2 **訂正あり**
- display result.ppm &

A            A'            B            B'



Shin Yoshizawa: shin@riken.jp

**授業評価アンケート**

- ✓ 授業評価アンケートを取りに来てください!
- ✓ 今日最後の10分は授業評価アンケートを書いてください。
- ✓ 科目名: **画像情報処理論及び演習I**
- ✓ 授業担当者: **吉澤 信**
- ✓ 「自由記入欄」は要望・意見があれば書いてください、そこだけは大学から後で私に回ってくるので、誰が書いたか分からない様にしてください。
- ✓ 後期の授業で授業の進め方 & 講義の内容でやってほしい事等があれば書いてください。
- ✓ 集めたら3階の第二共同研究室まで提出をお願いします。

Shin Yoshizawa: shin@riken.jp

**来週の予定**

[www.riken.jp/briect/Yoshizawa/Lectures/index.html](http://www.riken.jp/briect/Yoshizawa/Lectures/index.html)

- ① 13回~15回: 圧縮・周波数分解・符号化・ファイルI/O
- ② 演習.