機械学習を用いた 画像特徴量変換

小林 匠（コバヤシ タクミ）
国立研究開発法人 産業技術総合研究所
知能システム研究部門

特徴量変換とは

パターン認識における特徴量変換の位置付け

画像データ → 特徴抽出 → **特徴変換** → 判別・類別処理 → 識別処理 → 認識結果

SIFT・CNN
etc

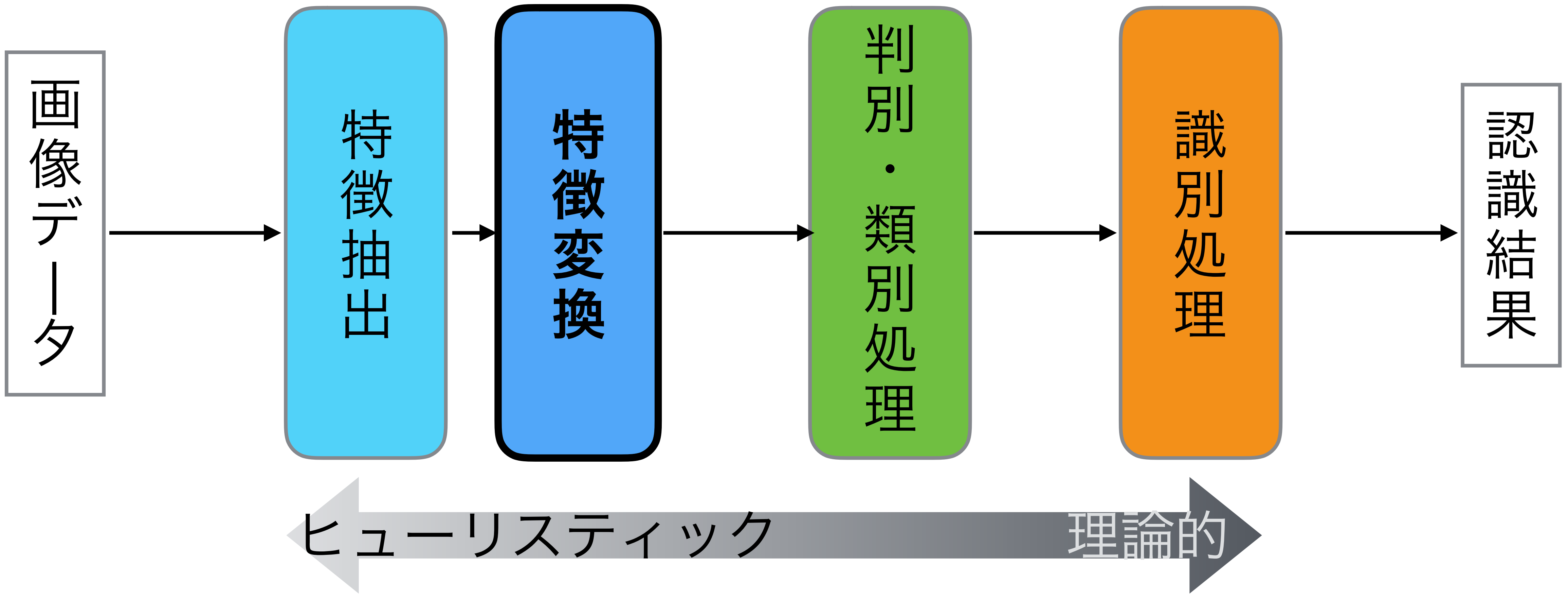
PCA・LDA
etc

SVM・kNN
etc

L2-normalization $\frac{\boldsymbol{x}}{\|\boldsymbol{x}\|_2}$
etc

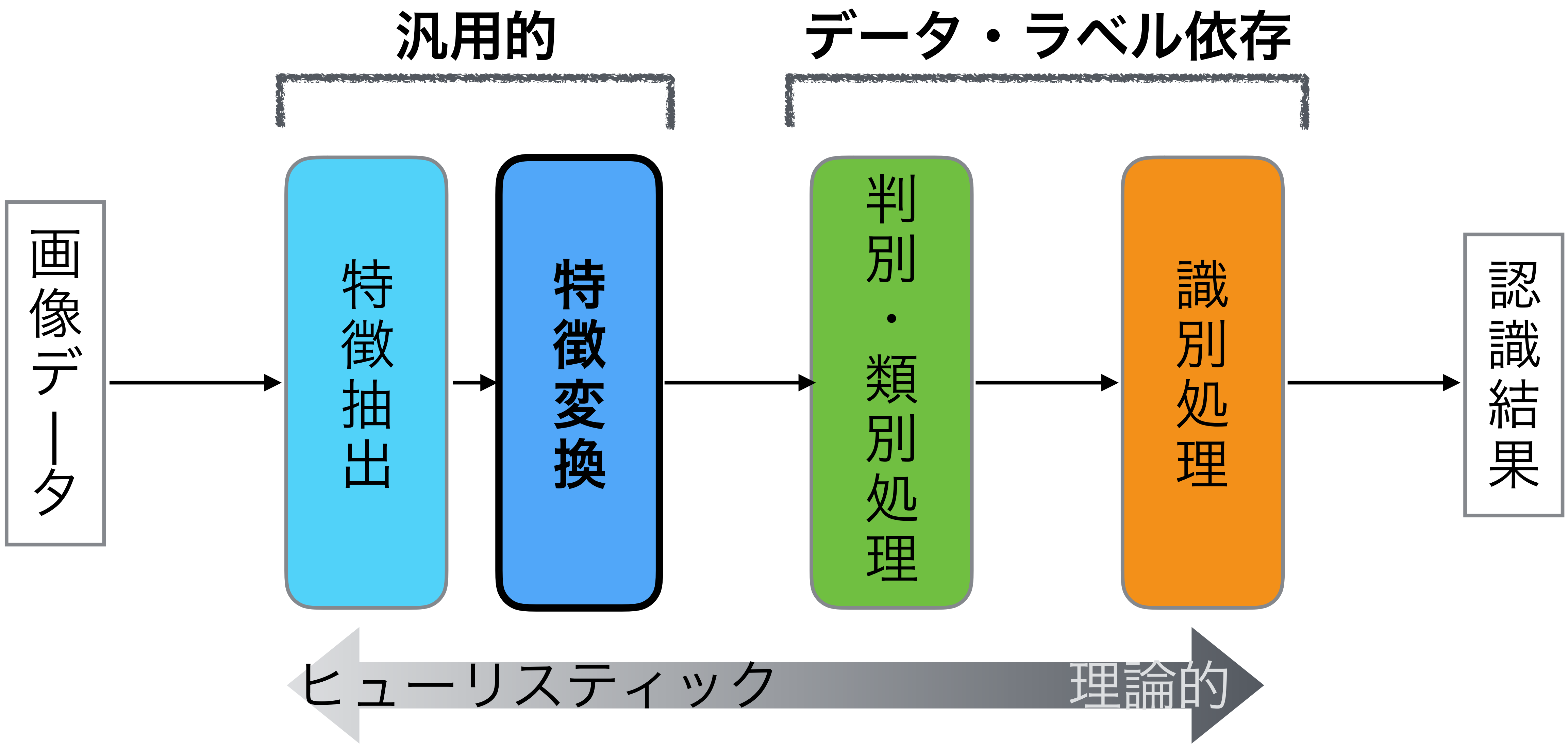
特徴量変換とは

パターン認識における特徴量変換の位置付け



特徴量変換とは

パターン認識における特徴量変換の位置付け



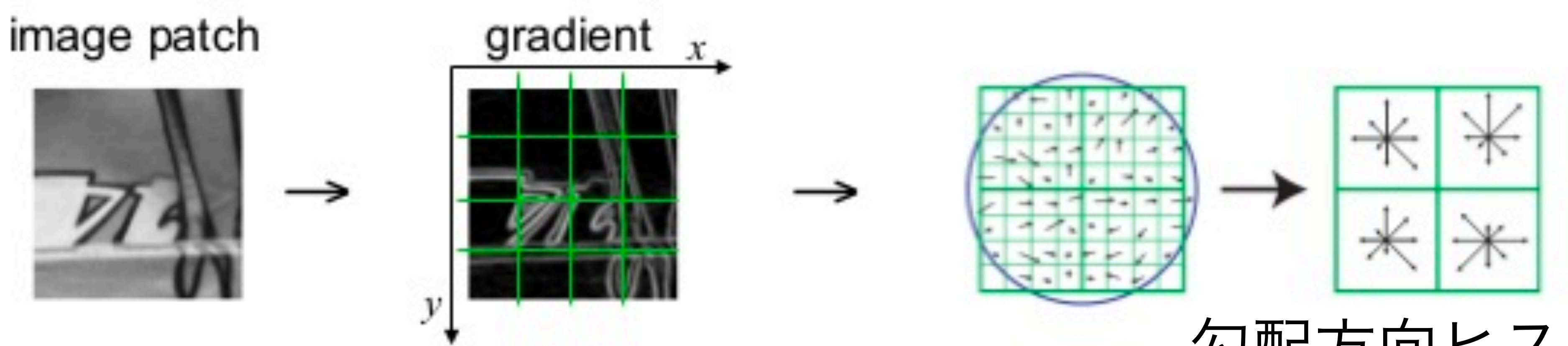
特徴変換： • 汎用性（データに依って処理を変えない）

• 弁別性（特徴の判別力を向上）

画像特徴量

人為的特徴量 = Hand-crafted特徴

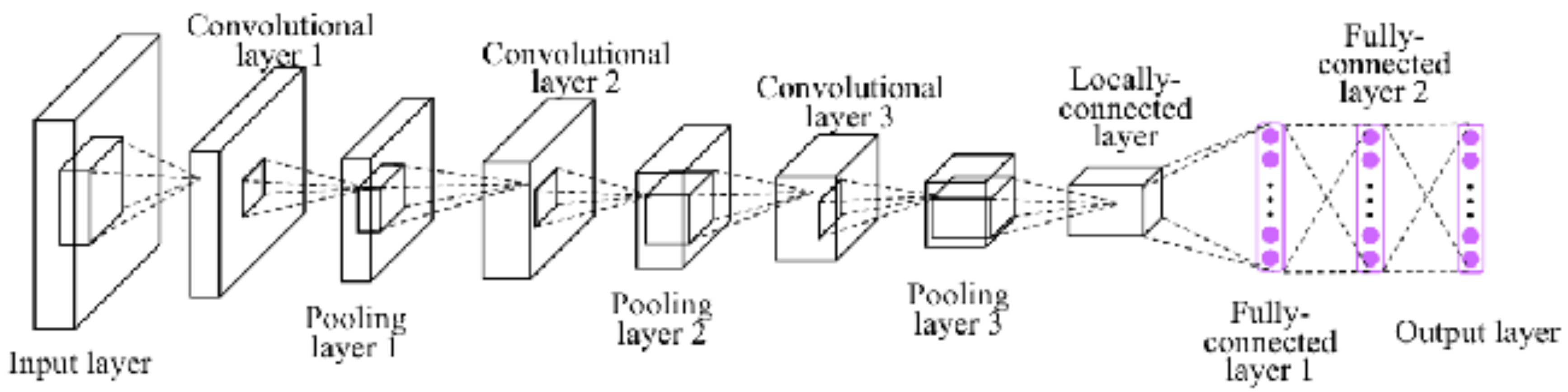
例：



勾配方向ヒストグラム
(SIFT)

学習的特徴量 = 畳み込みニューラルネットワーク(CNN)特徴

例：



転移された（pre-trained on ImageNet）CNNモデルを用いる

画像特徴量の変換

人為的特徴量 = Hand-crafted特徴

T. Kobayashi,
“**Structured Feature Similarity with Explicit Feature Map**”,
CVPR2016

学習的特徴量 = 畳み込みニューラルネット(CNN)特徴

T. Kobayashi,
“**Learning Additive Kernel For Feature Transformation and Its Application to CNN Features**”,
BMVC2016

科研費「パターン認識のための特徴量変換に関する研究」15K00261

人為的特徴量の変換

距離尺度

主にヒストグラム特徴に対する手法が提案されてきた

- χ^2(カイ２乗) -distance
- Earth Mover’s distance（EMD）
- Faster EMD （高速版EMD）
- SiftDist（SIFTでのEMD）
- Diffusion distance（ヒストグラム上の拡散過程）

ここでは特徴の物理的構造に着目した特徴変換を考える

ポイント

1. **特徴量のテンソル構造**
2. **SSIMに基づく距離尺度**

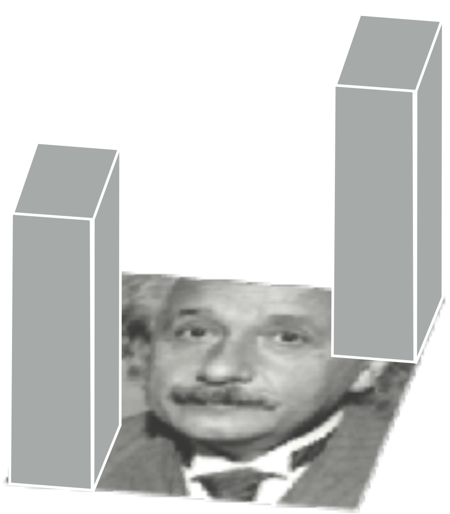
特徴量のテンソル構造（1）

画像特徴量は多くの場合にテンソル構造を内包する

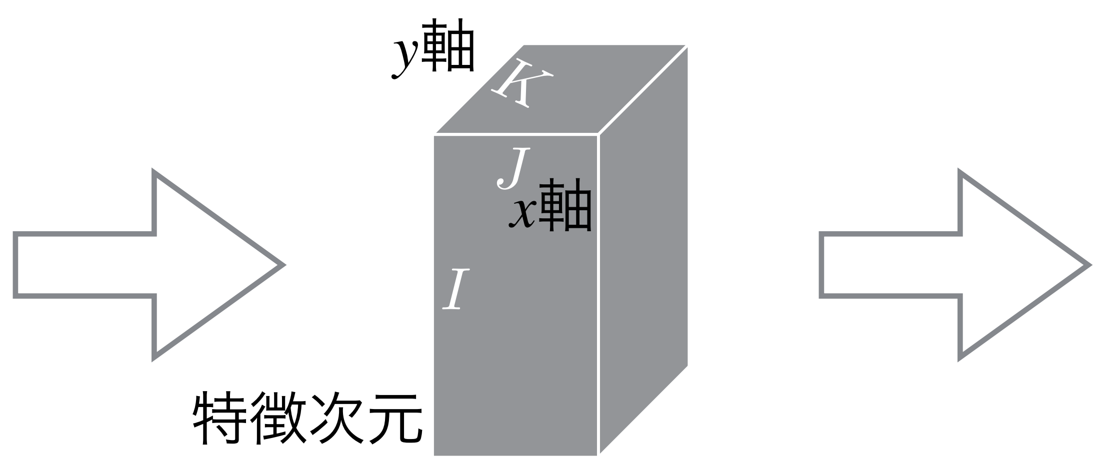
画像（x×y: 2 次元）から特徴量（1 次元）を抽出

→ ３階テンソル（x×y×特徴）

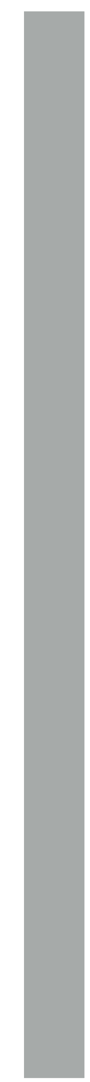
→ （従来は）ベクトル



画像の局所領域から
特徴抽出



３階テンソル



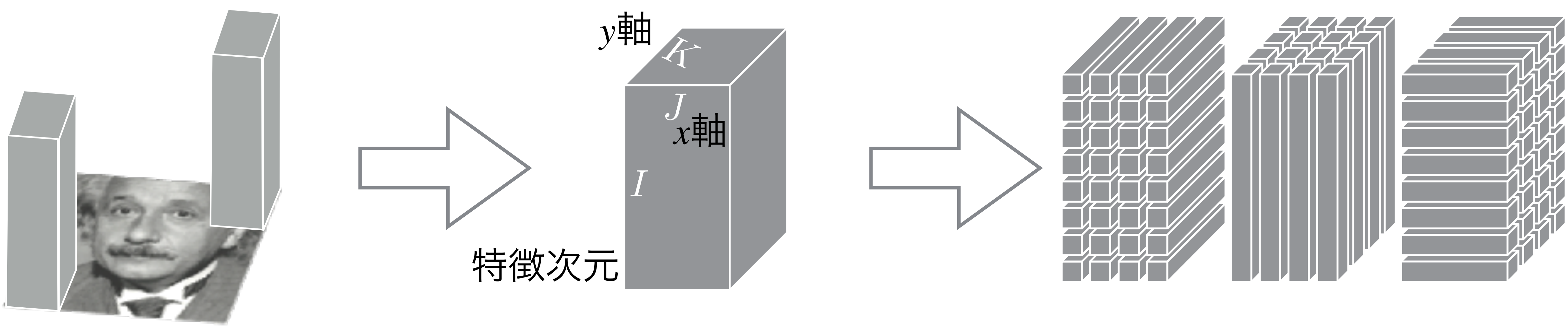
高次元ベクトル

特徴量のテンソル構造（2）

テンソル構造を保持した表現を考える

テンソルの各軸（x,y,特徴）に沿った特徴束を一つの単位として、そこでの距離尺度を考える

- 特徴束を大きくしすぎると、元のベクトル表現
- 特徴束を小さくしすぎると、特徴要素１個ずつ



画像の局所領域から特徴抽出

３階テンソル

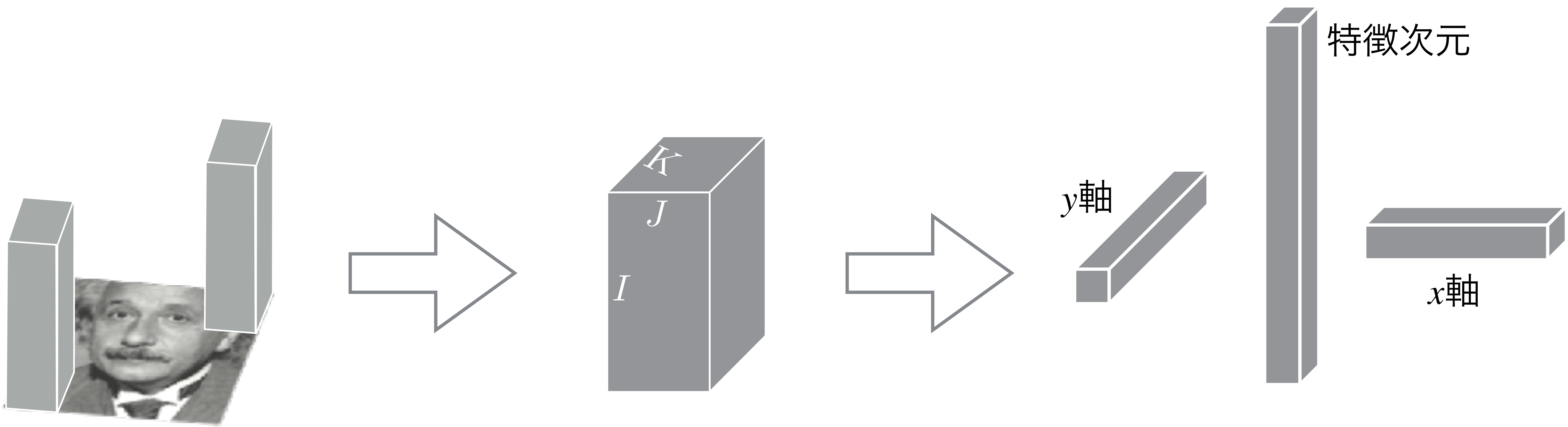
３軸それぞれに沿った特徴束

特徴量のテンソル構造（2）

テンソル構造を保持した表現を考える

テンソルの各軸（x,y,特徴）に沿った特徴束を一つの単位として、そこでの距離尺度を考える

- 特徴束を大きくしすぎると、元のベクトル表現
- 特徴束を小さくしすぎると、特徴要素１個ずつ



画像の局所領域から特徴抽出

３階テンソル

３軸それぞれに沿った特徴束（＝小さいベクトル）

SSIMに基づく距離尺度（1）

Structural **S**imilarity **I**ndex **M**easure（SSIM）

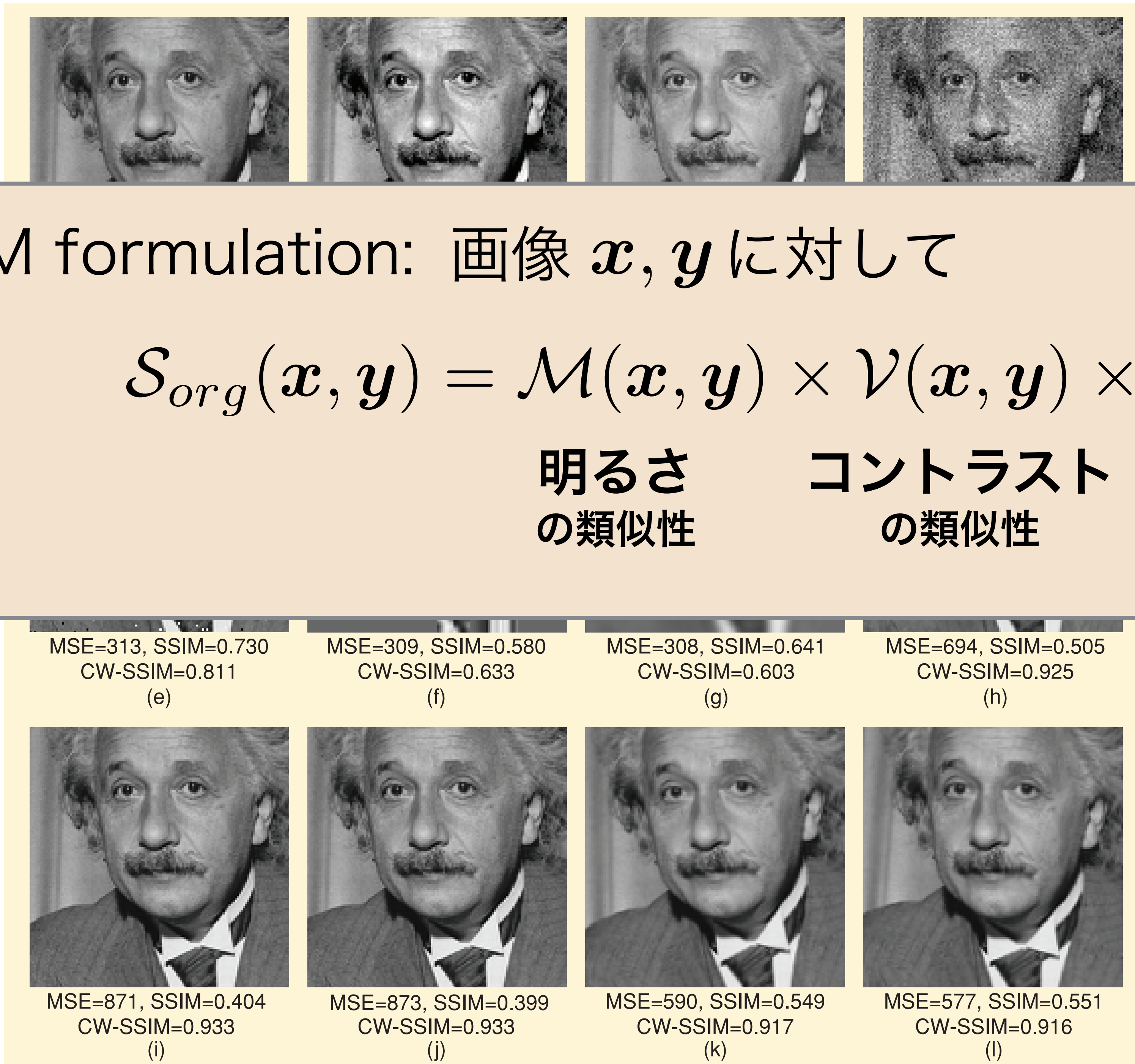
画像の品質評価のための定量的指標として用いられる

Z. Wang and A. C. Bovik. Mean squared error: love it or leave it? - a new look at signal fidelity measures. *IEEE Signal Processing Magazine*, 26(1):98–117, January 2009

SSIMに基づく距離尺度（1）

Structural **S**imilarity **I**ndex **M**easure（SSIM）

画像の品質評価のための定量的指標として用いられる



SSIM formulation: 画像 $\boldsymbol{x}, \boldsymbol{y}$ に対して

$$\mathcal{S}_{org}(\boldsymbol{x}, \boldsymbol{y}) = \mathcal{M}(\boldsymbol{x}, \boldsymbol{y}) \times \mathcal{V}(\boldsymbol{x}, \boldsymbol{y}) \times \mathcal{C}(\boldsymbol{x}, \boldsymbol{y})$$

明るさ の類似性　**コントラスト の類似性**　**構造パターン の類似性**

Z. Wang and A. C. Bovik. Mean squared error: love it or leave it? - a new look at signal fidelity measures. *IEEE Signal Processing Magazine*, 26(1):98–117, January 2009

SSIMに基づく距離尺度（2）

特徴束においてSSIMを距離尺度として採用

特徴束 $\boldsymbol{x}$ $\boldsymbol{y}$ に対して

$$\mathcal{S}_{org}(\boldsymbol{x},\boldsymbol{y}) = \mathcal{M}(\boldsymbol{x},\boldsymbol{y}) \times \mathcal{V}(\boldsymbol{x},\boldsymbol{y}) \times \mathcal{C}(\boldsymbol{x},\boldsymbol{y})$$

明るさ　コントラスト　構造パターン

相関係数

$$\mathcal{C}(\boldsymbol{x},\boldsymbol{y}) = \frac{(\boldsymbol{x}-\mathrm{u}(\boldsymbol{x}))^\top(\boldsymbol{y}-\mathrm{u}(\boldsymbol{y}))}{\|\boldsymbol{x}-\mathrm{u}(\boldsymbol{x})\|_2\|\boldsymbol{y}-\mathrm{u}(\boldsymbol{y})\|_2},$$

$\mathrm{k}(\mathrm{u}(\boldsymbol{x}),\mathrm{u}(\boldsymbol{y}))$, $\mathrm{k}(\mathrm{q}(\boldsymbol{x}),\mathrm{q}(\boldsymbol{y}))$, where **類似度関数** $\mathrm{k}(a,b) = \dfrac{2ab}{a^2+b^2}$

平均値の類似度　標準偏差の類似度

特徴変動への頑健性を高めるため乗法的表現から**加法的表現**へ

- 上記の元表現 $\mathcal{S}_{org}$ は、$\mathcal{M}$, $\mathcal{V}$, $\mathcal{C}$ のわずかな変化にも敏感

乗法的： $\mathcal{S}_{org} = \mathcal{M} \times \mathcal{V} \times \mathcal{C}$

加法的： $\mathcal{S}_{add} = w_\mathcal{M}\mathcal{M} + w_\mathcal{V}\mathcal{V} + w_\mathcal{C}\mathcal{C}$

$(w_\mathcal{M}, w_\mathcal{V}, w_\mathcal{C}) = (2,2,1)$

	$\mathcal{M}$	$\mathcal{V}$	$\mathcal{C}$
non-neg. feat.	[0, +1]	[0, +1]	[-1, +1]
real feat.	[-1, +1]	[0, +1]	[-1, +1]

value range

$(w_\mathcal{M}, w_\mathcal{V}, w_\mathcal{C}) = (1,2,1)$

効率的表現 ⇔ 特徴"変換"の実現（1）

類似度 $\mathcal{S}$ を陽写像（explicit map）の内積へ展開

- 線形表現になり、高速演算かつ線形識別器を構築可能

$$\mathcal{S}(\boldsymbol{x},\boldsymbol{y}) = w_{\mathcal{M}}\mathcal{M}(\boldsymbol{x},\boldsymbol{y}) + w_{\mathcal{V}}\mathcal{V}(\boldsymbol{x},\boldsymbol{y}) + w_{\mathcal{C}}\mathcal{C}(\boldsymbol{x},\boldsymbol{y})$$

$$= \boldsymbol{\phi}(\boldsymbol{x})^{\top}\boldsymbol{\phi}(\boldsymbol{y})$$

陽写像（explicit map）$\in \mathbb{R}^d$

効率的表現 ⇔ 特徴"変換"の実現（1）

類似度 $\mathcal{S}$ を陽写像（explicit map）の内積へ展開

- 線形表現になり、高速演算かつ線形識別器を構築可能

$$\mathcal{S}(\boldsymbol{x},\boldsymbol{y}) = w_{\mathcal{M}}\mathcal{M}(\boldsymbol{x},\boldsymbol{y}) + w_{\mathcal{V}}\mathcal{V}(\boldsymbol{x},\boldsymbol{y}) + w_{\mathcal{C}}\mathcal{C}(\boldsymbol{x},\boldsymbol{y})$$

$$= \underbrace{\phi}_{}(\boldsymbol{x})^{\top}\phi(\boldsymbol{y})$$

陽写像（explicit map）$\in \mathbb{R}^d$

$$\phi = \sqrt{w_{\mathcal{M}}}\phi_{\mathcal{M}} \oplus \sqrt{w_{\mathcal{V}}}\phi_{\mathcal{C}} \oplus \sqrt{w_{\mathcal{V}}}\phi_{\mathcal{C}}$$

直和

効率的表現 ⇔ 特徴"変換"の実現（1）

類似度 $\mathcal{S}$ を陽写像（explicit map）の内積へ展開

- 線形表現になり、高速演算かつ線形識別器を構築可能

$$\mathcal{S}(\boldsymbol{x},\boldsymbol{y}) = w_{\mathcal{M}}\mathcal{M}(\boldsymbol{x},\boldsymbol{y}) + w_{\mathcal{V}}\mathcal{V}(\boldsymbol{x},\boldsymbol{y}) + w_{\mathcal{C}}\mathcal{C}(\boldsymbol{x},\boldsymbol{y})$$

$$= \phi(\boldsymbol{x})^\top \phi(\boldsymbol{y})$$

陽写像（explicit map）$\in \mathbb{R}^d$

$$\phi = \sqrt{w_{\mathcal{M}}}\phi_{\mathcal{M}} \oplus \sqrt{w_{\mathcal{V}}}\phi_{\mathcal{C}} \oplus \sqrt{w_{\mathcal{V}}}\phi_{\mathcal{C}}$$

直和

✔ $$\mathcal{C}(\boldsymbol{x},\boldsymbol{y}) = \boxed{\frac{(\boldsymbol{x}-\mathrm{u}(\boldsymbol{x}))}{\|\boldsymbol{x}-\mathrm{u}(\boldsymbol{x})\|_2}}^\top \boxed{\frac{(\boldsymbol{y}-\mathrm{u}(\boldsymbol{y}))}{\|\boldsymbol{y}-\mathrm{u}(\boldsymbol{y})\|_2}},$$

? $$\mathcal{M}(\boldsymbol{x},\boldsymbol{y}) = \mathtt{k}(\mathrm{u}(\boldsymbol{x}),\mathrm{u}(\boldsymbol{y})),$$
$$\mathcal{V}(\boldsymbol{x},\boldsymbol{y}) = \mathtt{k}(\mathrm{q}(\boldsymbol{x}),\mathrm{q}(\boldsymbol{y})),$$

$$\mathtt{k}(a,b) = \frac{2ab}{a^2+b^2}$$

効率的表現 ⇔ 特徴"変換"の実現（2）

類似度関数 k を陽写像（explicit map）の内積へ展開

$$\mathrm{k}(a,b)=\frac{2ab}{a^2+b^2}=\frac{2\mathrm{sgn}(ab)}{\left|\frac{a}{b}\right|+\left|\frac{b}{a}\right|}=\frac{2\mathrm{sgn}(ab)}{e^{-\omega}+e^{\omega}}=\mathrm{sgn}(ab)\mathrm{sech}(\omega)$$

where $\omega = \log\left|\frac{b}{a}\right|$

関数sechのフーリエ展開に基いて

$$\begin{aligned}\mathrm{k}(a,b)=\mathrm{sgn}(ab)\mathrm{sech}(\omega)&=\mathrm{sgn}(ab)\int_{-\infty}^{\infty}e^{-i\omega\lambda}\kappa(\lambda)d\lambda\\&=\int_{-\infty}^{\infty}[\mathrm{sgn}(a)e^{-i\lambda\log|a|}\sqrt{\kappa(\lambda)}]^*[\mathrm{sgn}(b)e^{-i\lambda\log|b|}\sqrt{\kappa(\lambda)}]d\lambda,\end{aligned}$$

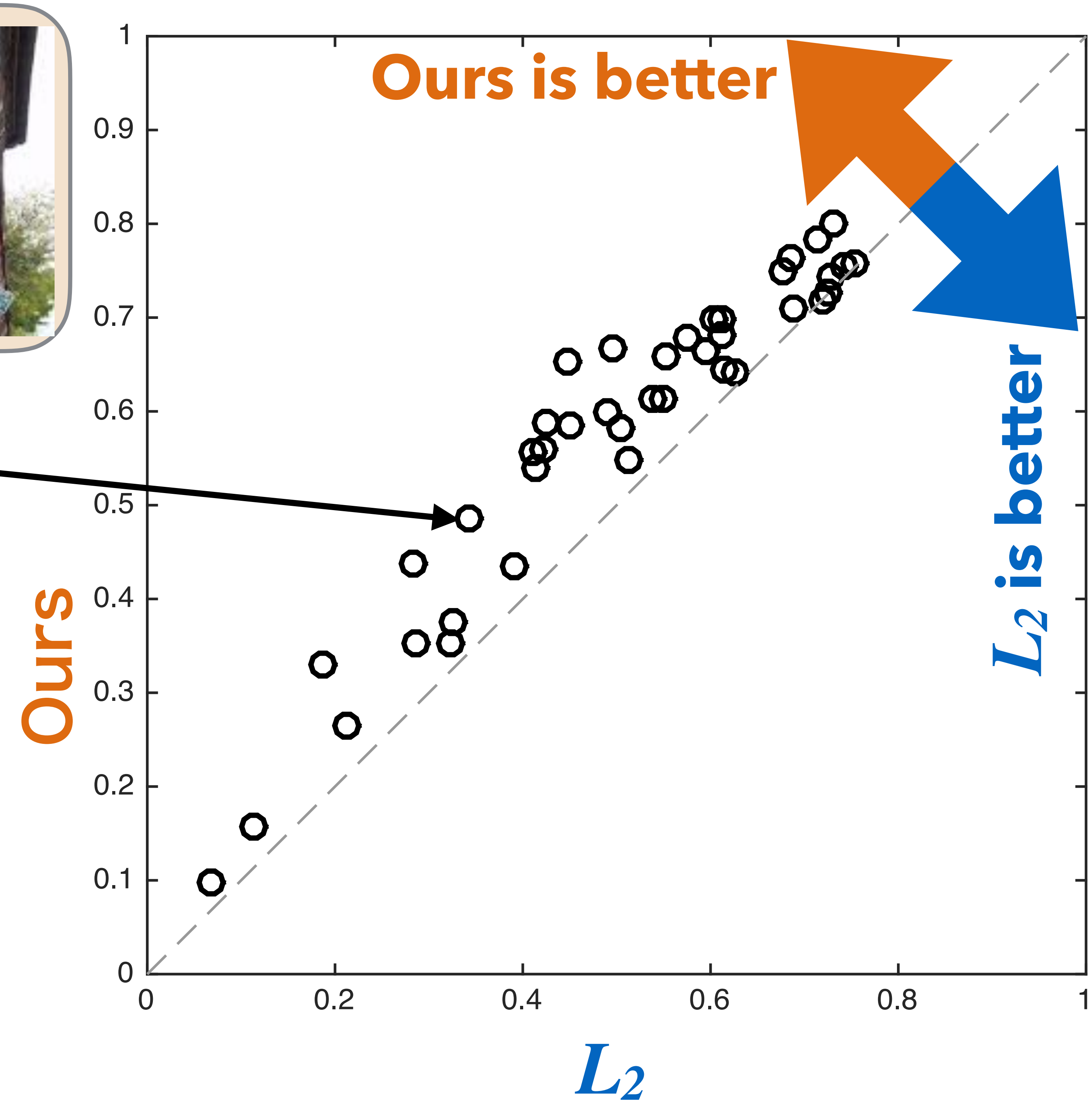
$$\Rightarrow\quad \phi_{\mathrm{k}}=\begin{bmatrix}[\![a=0]\!]\\ \tilde{\mathrm{g}}_k(\lambda;a)=\mathrm{sgn}(a)e^{-i\lambda\log|a|}\sqrt{\frac{1}{2}\mathrm{sech}\left(\frac{\pi\lambda}{2}\right)}\end{bmatrix}\in\mathbb{R}^{d_{\mathrm{k}}}$$

λを複数個サンプリング

実験結果

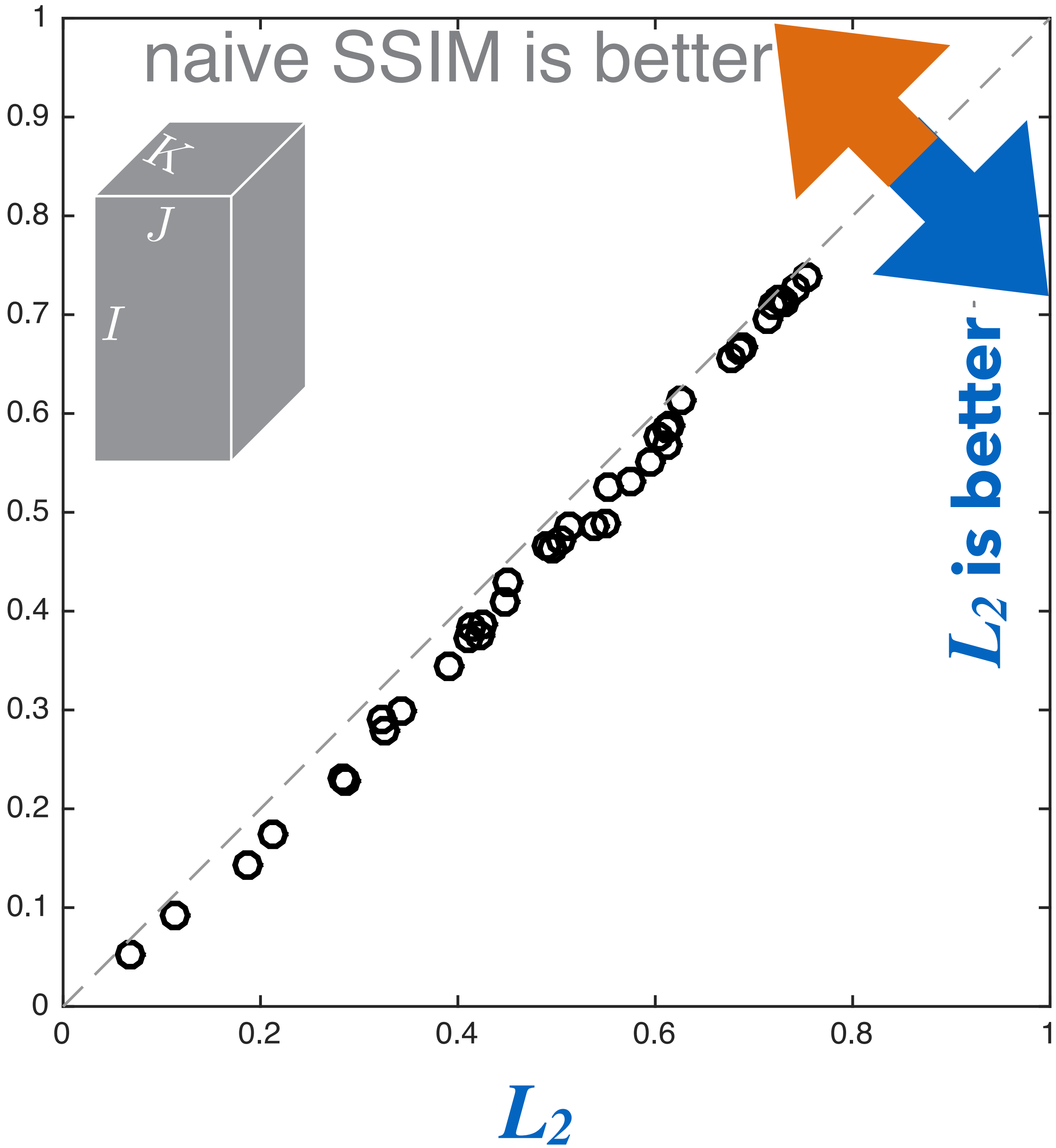
keypointマッチング＠INRIAデータセット

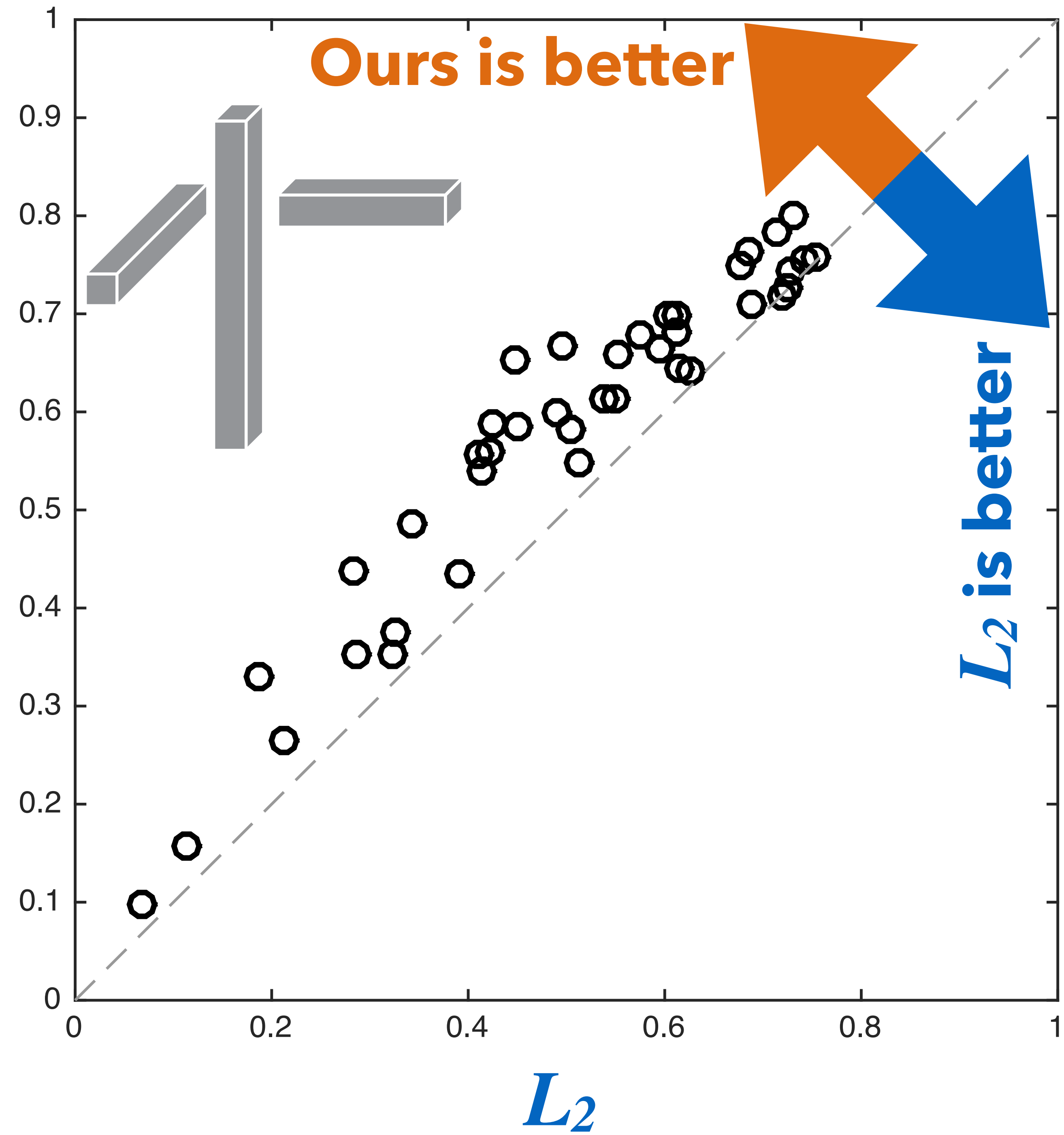
$$\begin{pmatrix} \text{Average Precision by Ours} \\ \text{Average Precision by } L_2 \end{pmatrix}$$



実験結果

keypointマッチング@INRIAデータセット

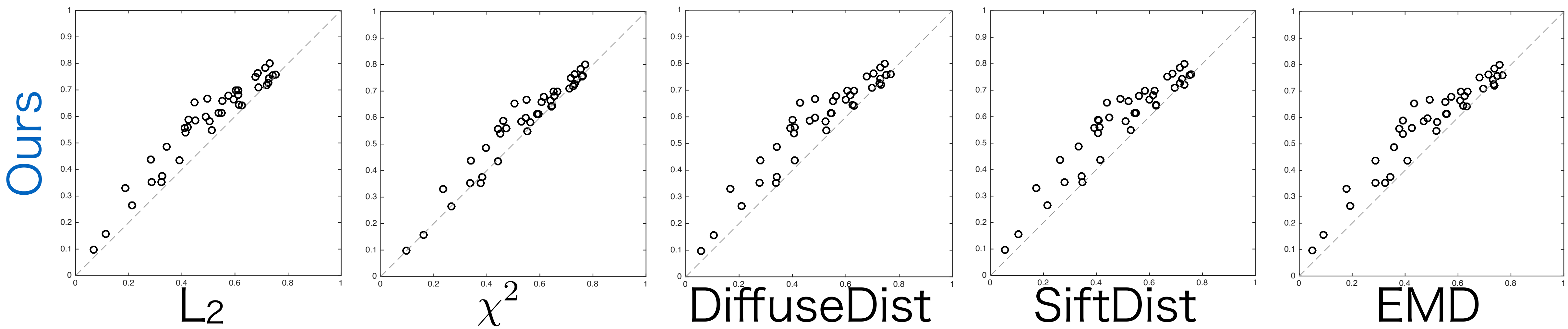




実験結果

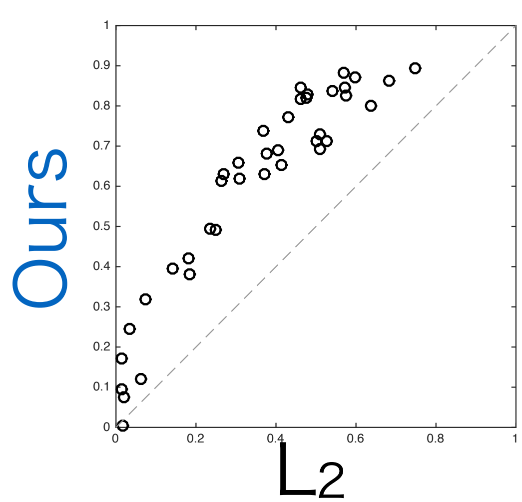
keypointマッチング＠INRIAデータセット

SIFT特徴量



その他色々な距離尺度

SURF特徴量



✓ 性能向上

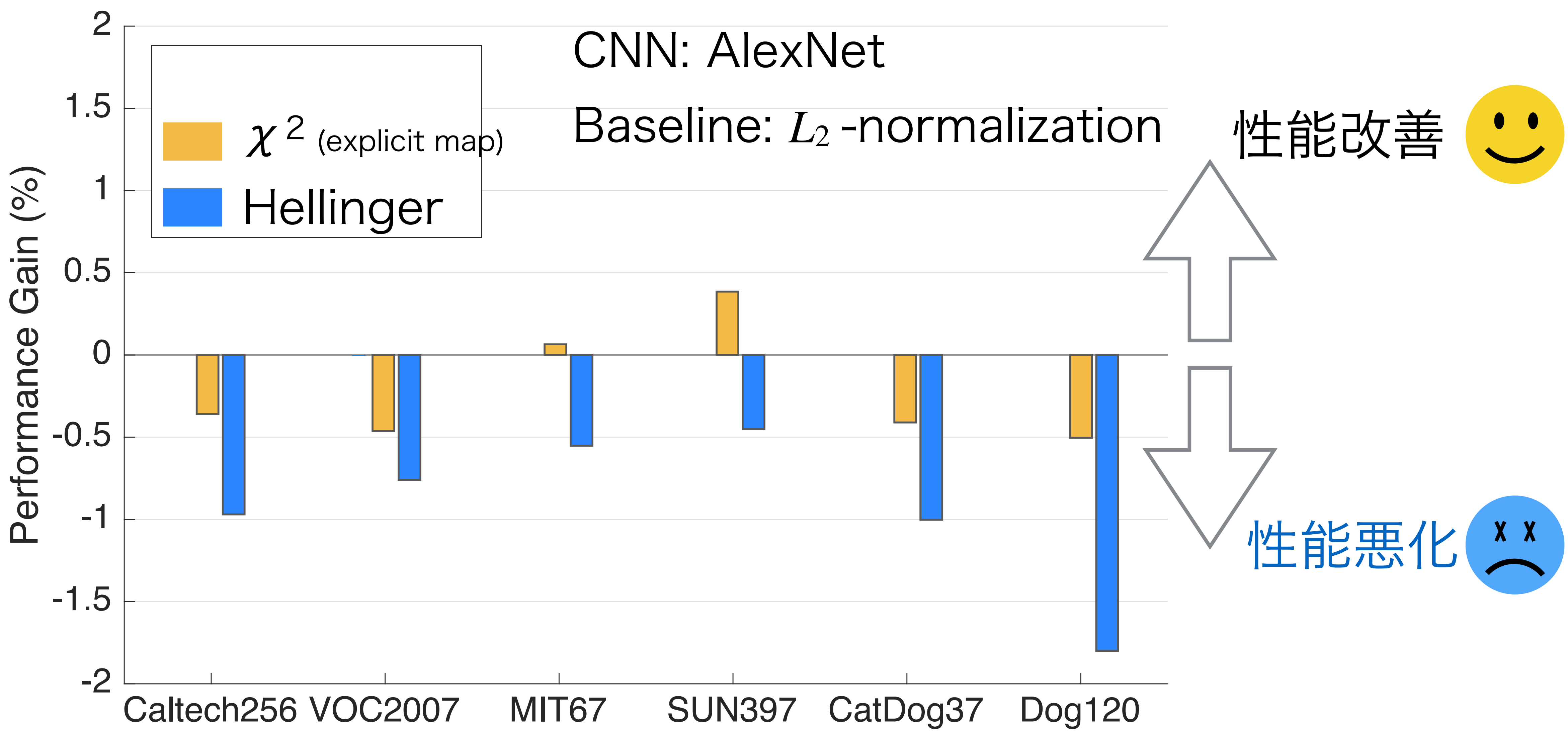
✓ 計算コスト小

学習的特徴量の変換

学習的特徴量では？

ヒストグラムなどの人為的特徴量では特徴変換はうまくいくが、学習的特徴量（CNN特徴）でもうまくいくか？

- 実はあまりうまくいかない…



なぜか？

距離尺度＝非線形（kernel）関数は、対象の特徴量の特性に沿って決めていた。

CNN特徴のような”よくわからない”特徴に対して強引に適用するのは困難であるらしい。

そこで、学習的特徴量に対する距離尺度も学習する

ポイント

1. **距離尺度のデータからの学習**
2. **弁別性・汎化性の高い変換方式**

加法的カーネル表現

加法的カーネルから出発

D次元特徴ベクトル $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^D$ に対して

加法的カーネル: $$\bar{\mathrm{k}}(\boldsymbol{x}, \boldsymbol{y}) = \sum_{i=1}^{D} \mathrm{k}(x_i, y_i),$$ 要素毎のカーネル関数の和

カーネル関数の陽写像展開：

$$\mathrm{k}(x,y) = \int \phi(\gamma;x)\phi(\gamma;y)d\lambda \approx \sum_{l=1}^{K} \phi(\gamma_l;x)\phi(\gamma_l;y)\Delta_l$$
$$= \boldsymbol{\phi}(x)^\top \boldsymbol{\phi}(y)$$

加法的カーネルの学習 → 関数 $\boldsymbol{\phi}(x)$ の学習

加法的カーネル表現の学習（1）

関数 $\boldsymbol{\phi}(x)$ を一から学習するのは困難。

そこで、フーリエ展開的に基底関数 $\mathtt{f}_m$ を導入して
関数 $\boldsymbol{\phi}(x)$ を表現する。

$$\boldsymbol{\phi}(x) = \boldsymbol{W}^\top \underbrace{[\mathtt{f}_1(x), \cdots, \mathtt{f}_M(x)]^\top}_{M\text{個の基底関数}} = \underbrace{\boldsymbol{W}^\top}_{\substack{\text{係数行列} \\ \mathbb{R}^{M \times K}}} \mathbf{f}(x)$$

基底関数が与えられているとすると

関数 $\boldsymbol{\phi}(x)$ の学習 → 係数行列 $\boldsymbol{W}$ の学習

基底関数については後述

加法的カーネル表現の学習（2）

このときカーネル関数は

$$\bar{\mathtt{k}}(\boldsymbol{x},\boldsymbol{y}) = \sum_{i=1}^{D} \mathbf{f}(x_i)^\top \boldsymbol{W}\boldsymbol{W}^\top \mathbf{f}(y_i) = \mathtt{tr}\{\boxed{\mathbf{F}(\boldsymbol{x})}^\top \boldsymbol{W}\boldsymbol{W}^\top \mathbf{F}(\boldsymbol{y})\},$$

where $\mathbf{F}(\boldsymbol{x}) = [\mathbf{f}(x_1), \cdots, \mathbf{f}(x_D)] \in \mathbb{R}^{M\times D}$.

要素毎の基底関数へのマッピング
を並べたもの=特徴行列

その識別関数は

bilinear形式: $\mathtt{tr}\{\boldsymbol{W}^\top \mathbf{F}(\boldsymbol{x})\boldsymbol{A}\} + b$

$\boldsymbol{W}$: 係数行列　$\mathbb{R}^{M\times K}$
$\boldsymbol{A}$: 識別重み行列　$\mathbb{R}^{D\times K}$

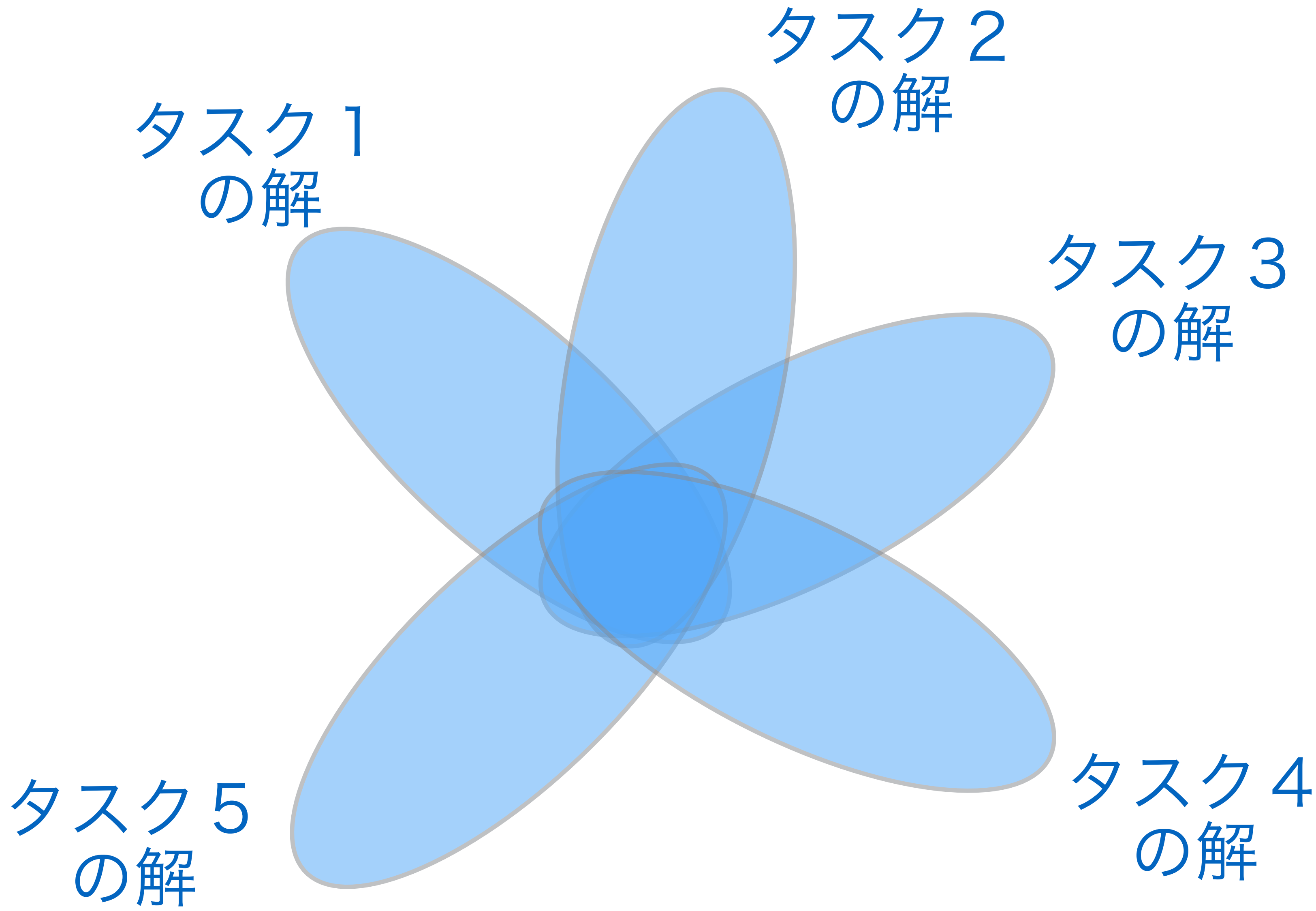
係数行列Wと識別重み行列Aの同時学習

加法的カーネル表現の学習（3）

汎化性と弁別性のトレードオフを考える必要がある

- 学習するデータに依存せず一定の解が欲しい

➡ 様々な学習タスクを用意し活用する

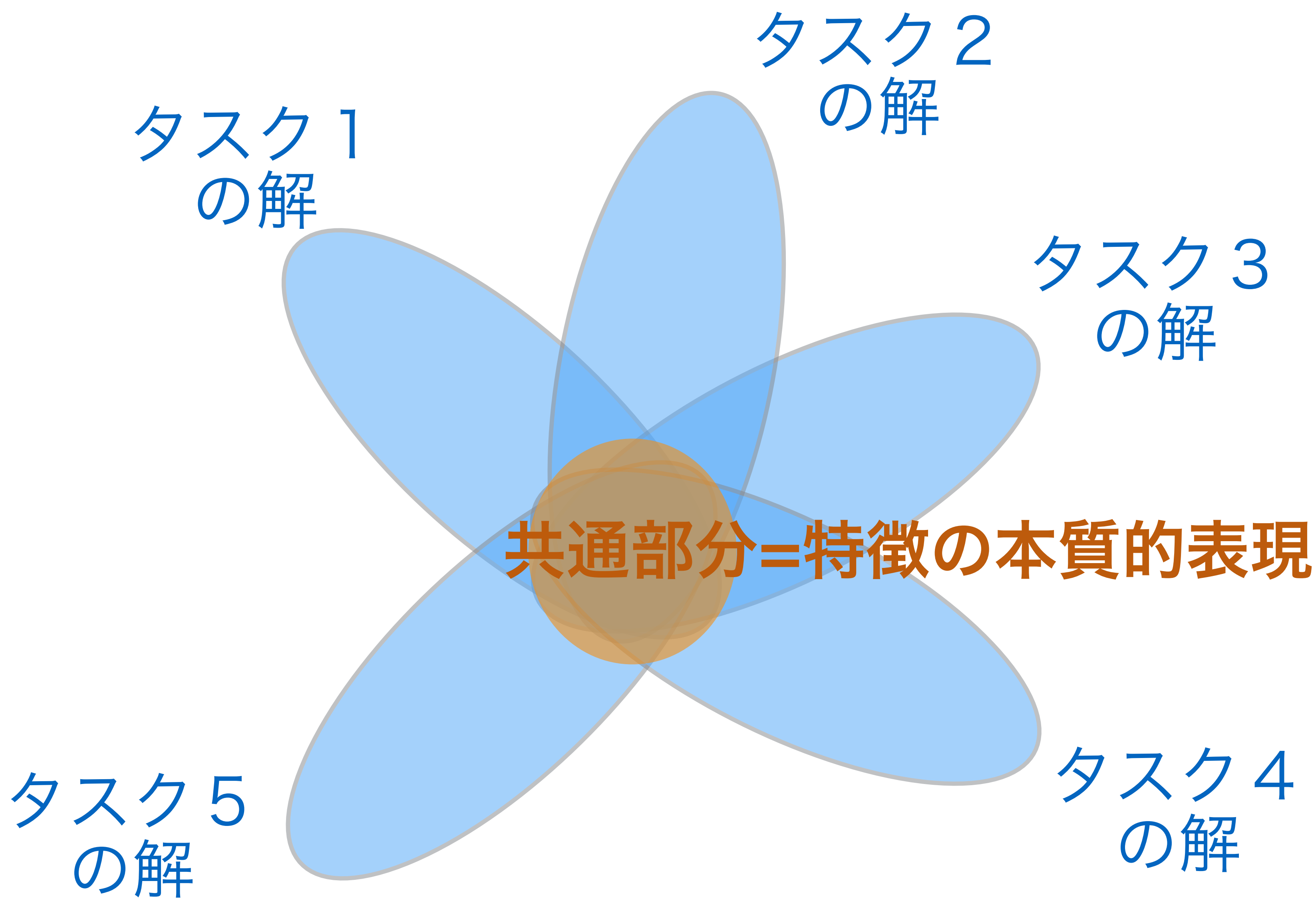


加法的カーネル表現の学習（3）

汎化性と弁別性のトレードオフを考える必要がある

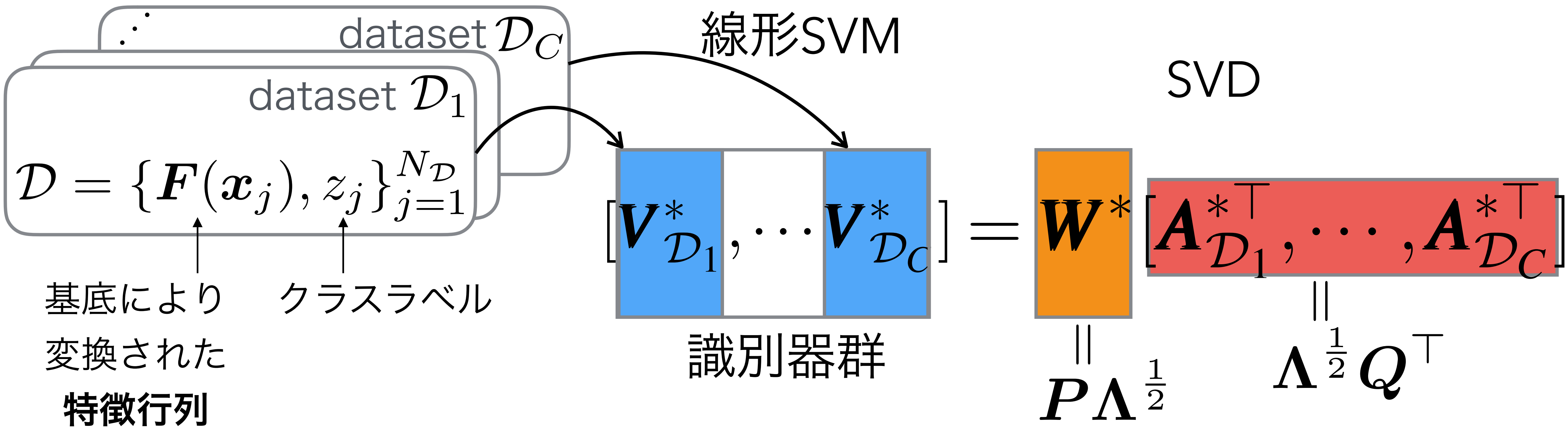
- 学習するデータに依存せず一定の解が欲しい

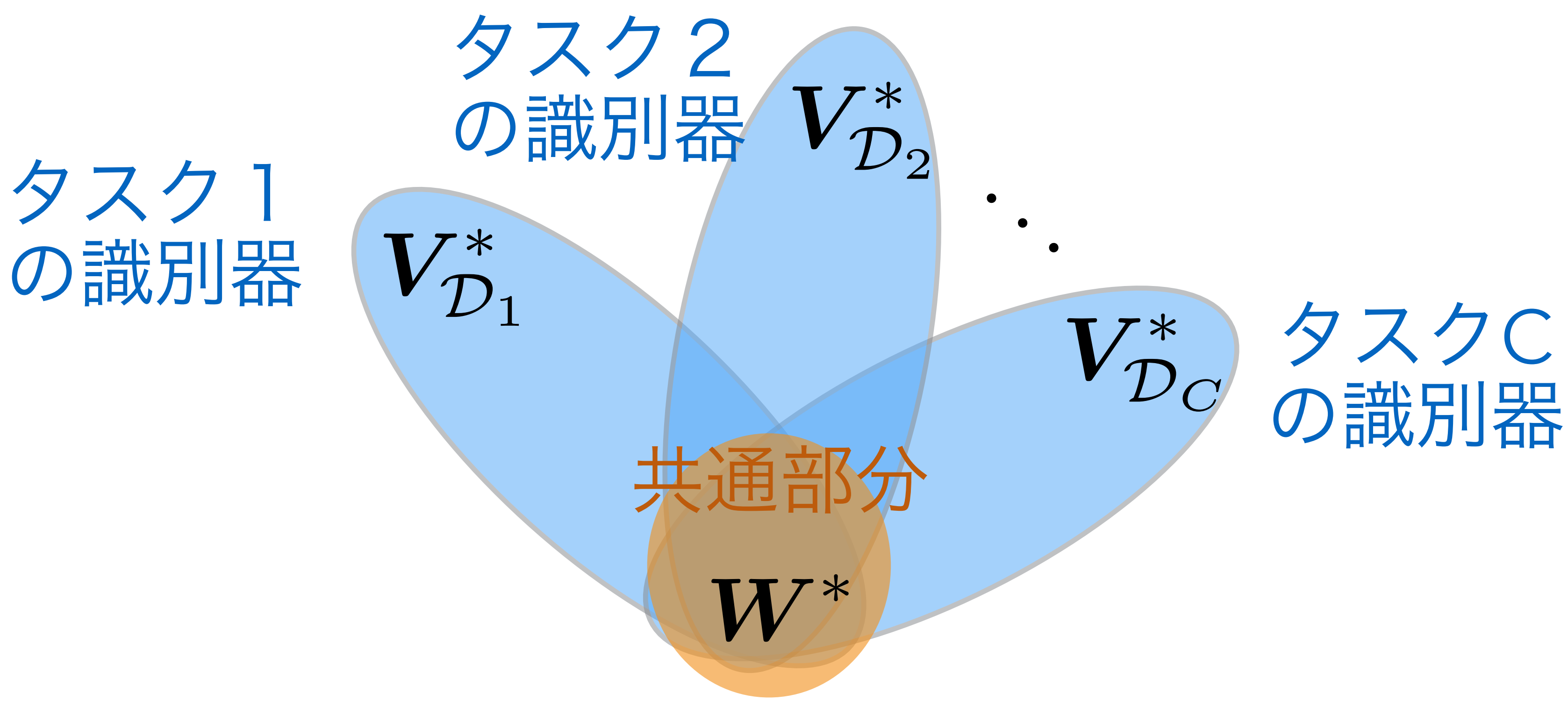
→ <u>様々な学習タスク</u>を用意し活用する



加法的カーネル表現の学習（4）

識別器の共通表現をSVDで抽出することで汎化性を高める





基底関数

フーリエ基底を考えるのが自然。

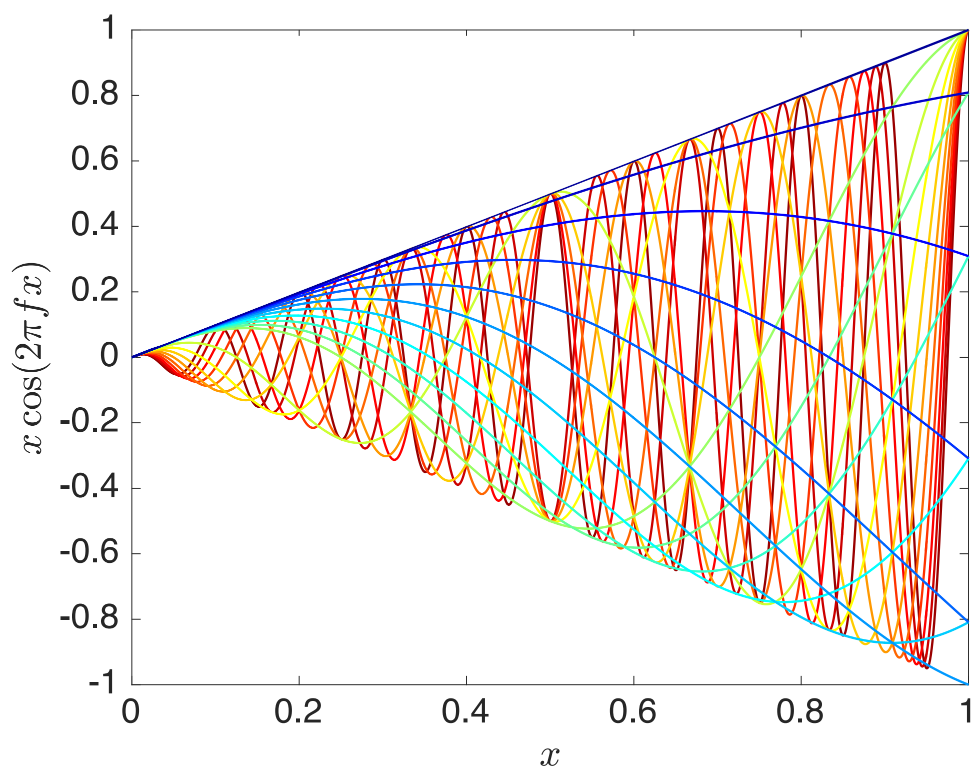
ここでは更に、特徴の根本的な性質を考慮

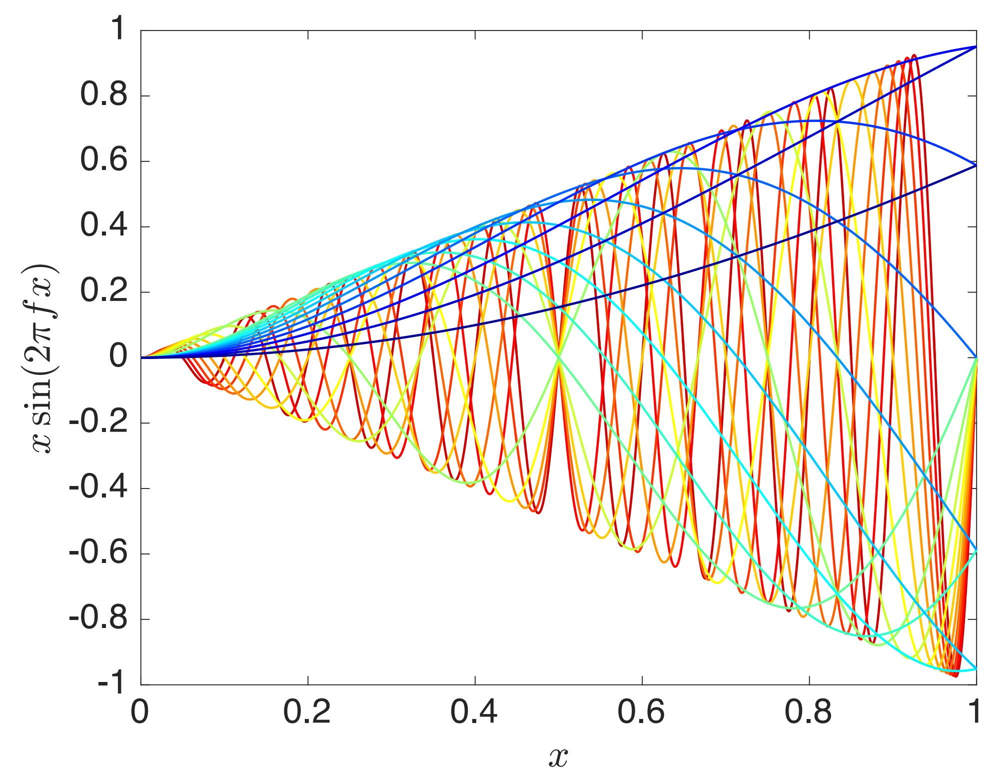
↓

（magnitudeの）大きい特徴値はより特徴的な情報を含む

$$\mathtt{f}_{2m-1}(x) = x\cos(2\pi\eta_m x), \quad \mathtt{f}_{2m}(x) = x\sin(2\pi\eta_m x),$$

where $\eta \in \{0, 0.1, \cdots, 0.9, 1, \cdots, 10\}$,

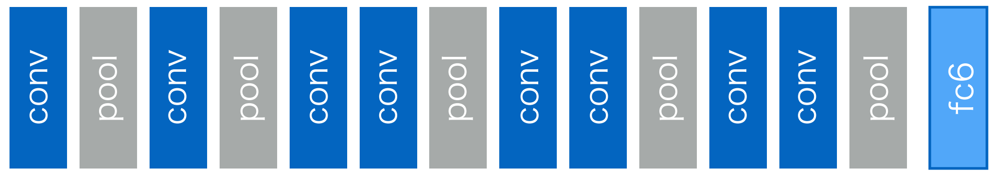




実験結果

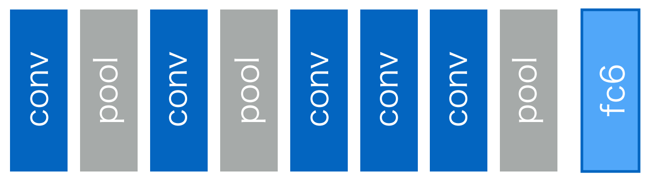
３種類のCNNモデル

C3D
[Tran, 2015]



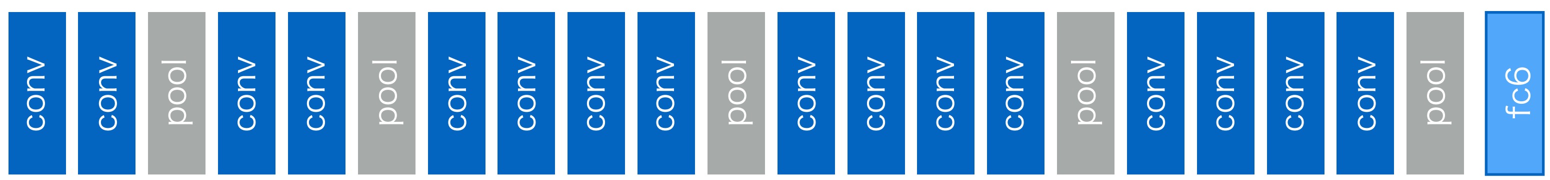
for ***Videos***
pretrained on sports1M

Alex
[Krizhevsky, 2012]



for ***Images***
pretrained on ImageNet

VGG
[Simonyan, 2014]



for ***Images***
pretrained on ImageNet

データセット

Video: HMDB51, UCF101/50, Hollywood2（動作認識）

Image: Caltech256, VOC2007, (物体認識)

MIT67, SUN397, (シーン認識)

CatDog37, Dog120 (詳細な種別認識)

Video Datasets

HMDB51

shoot gun

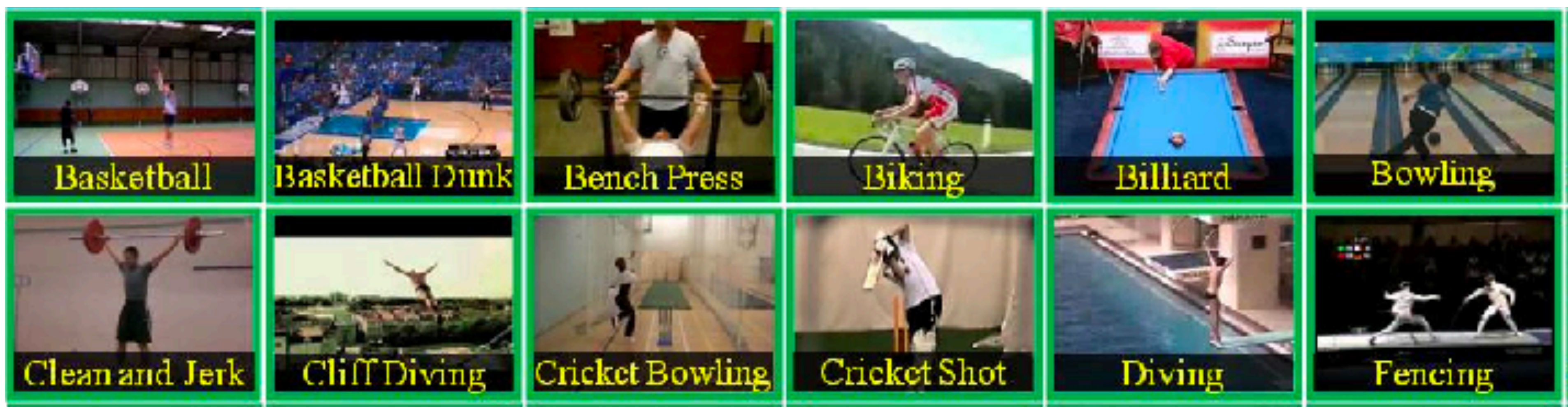
sit

situp

smile

smoke

UCF101/50



Hollywood2

Image Datasets

Caltech256 (物体)

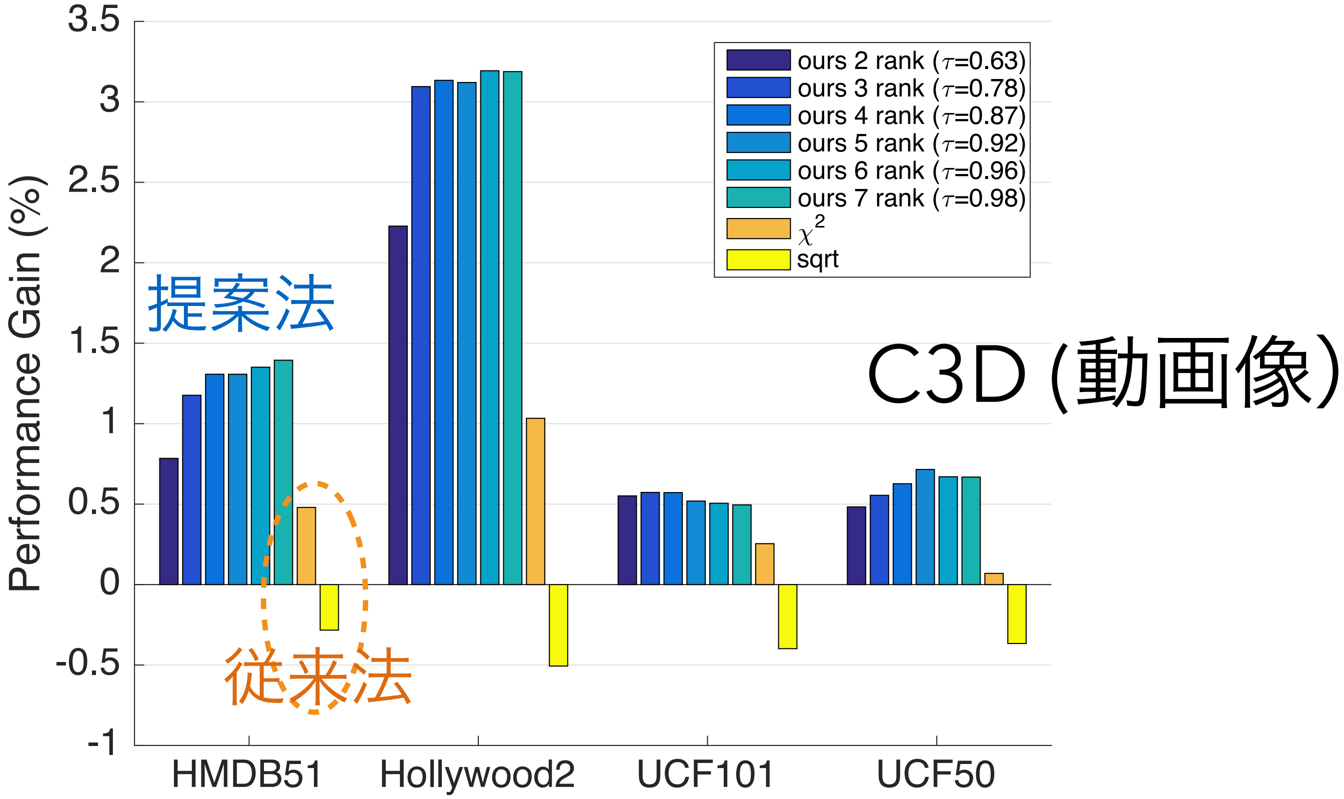
SUN397 (シーン)

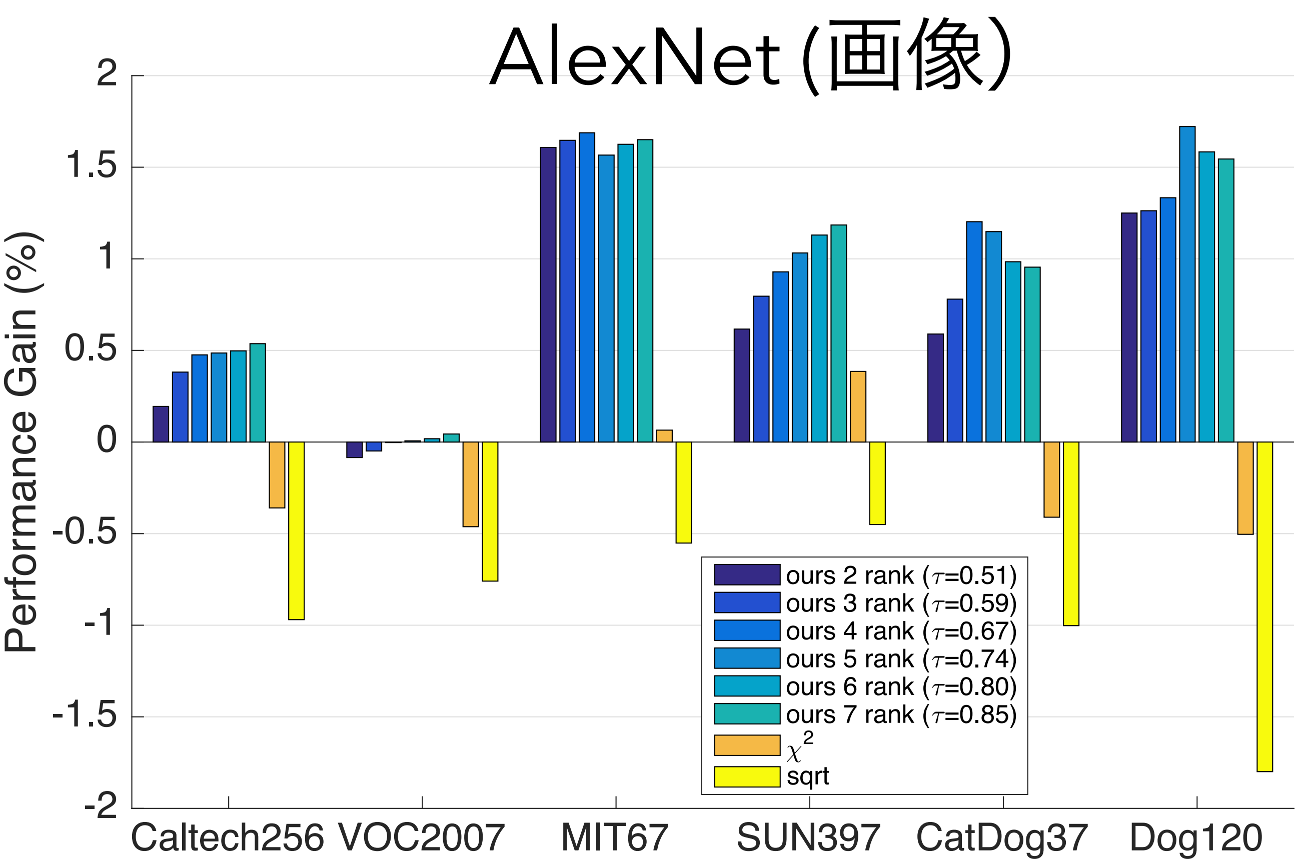
Dog120 (種別)

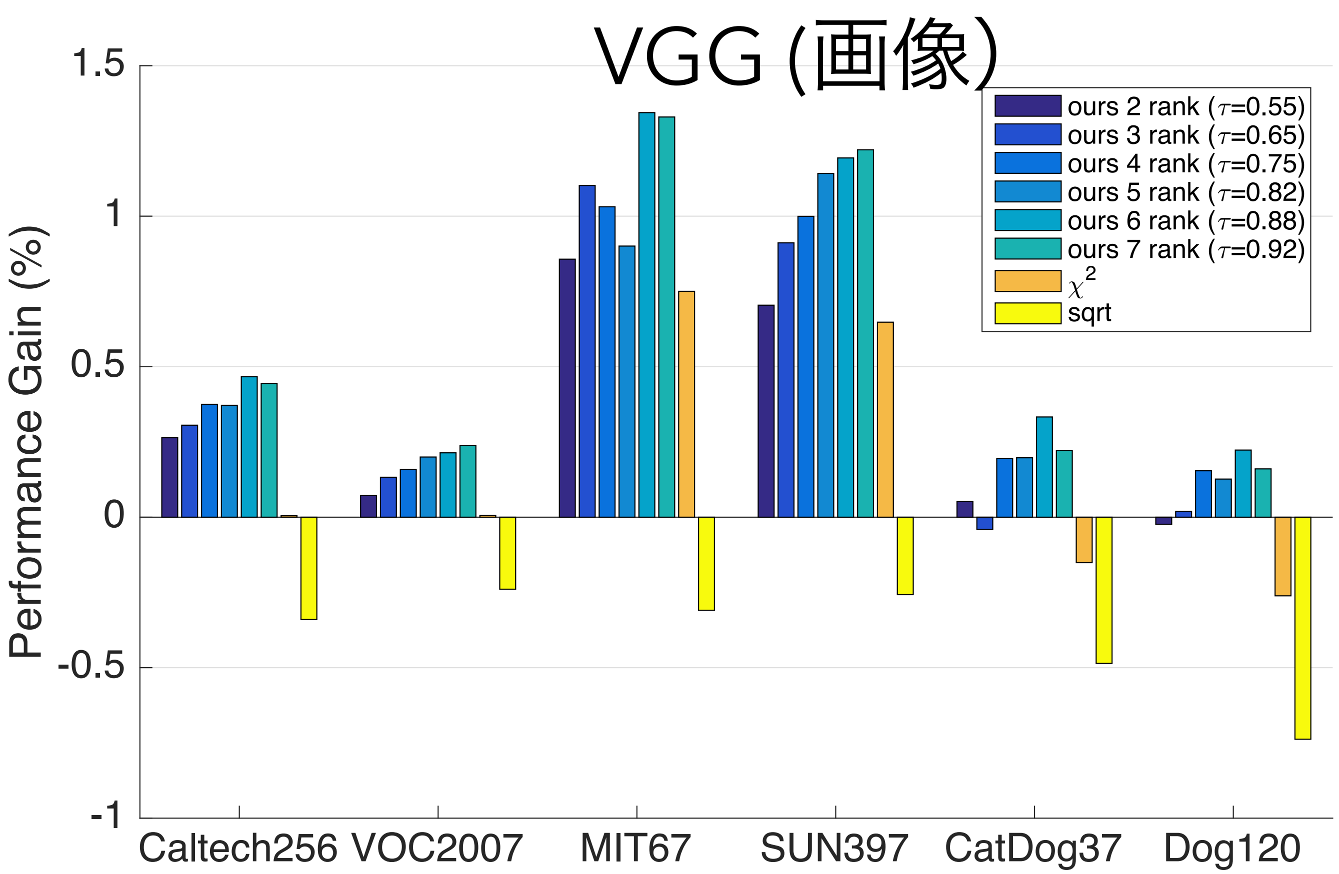
実験結果

baseline: L_2

- 提案法により性能向上

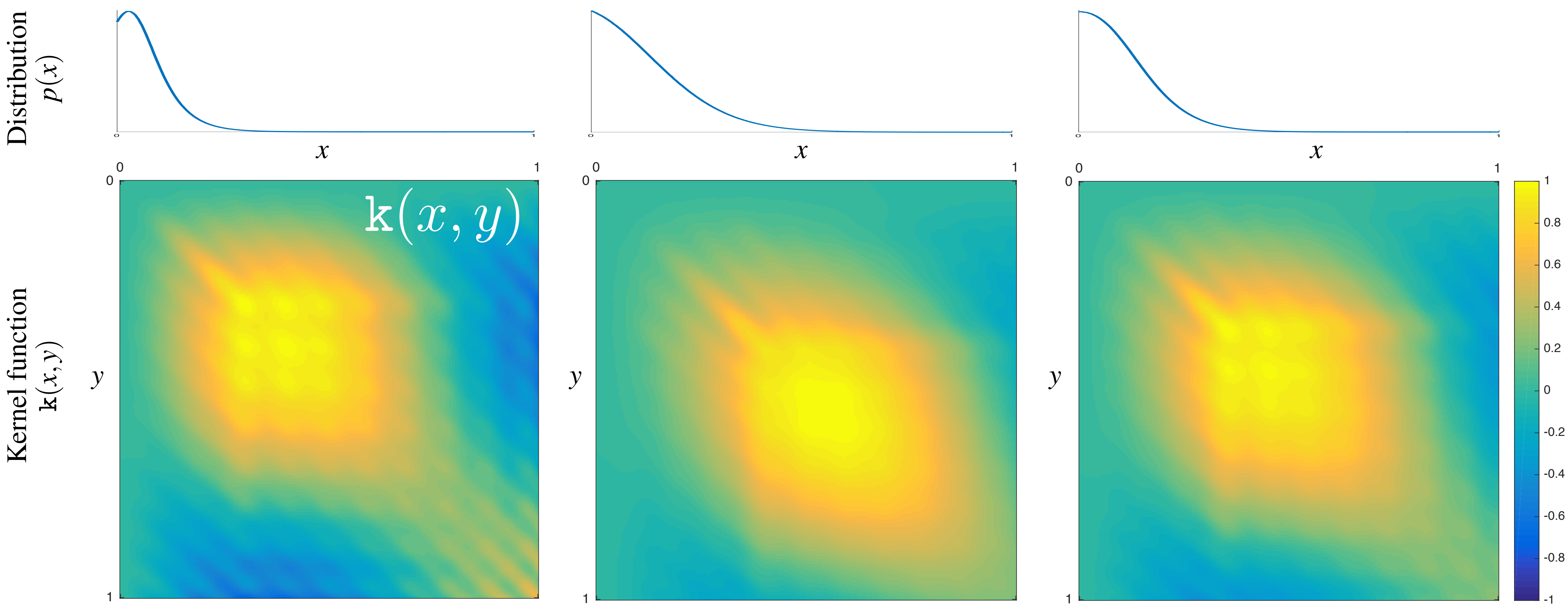






学習されたカーネル関数

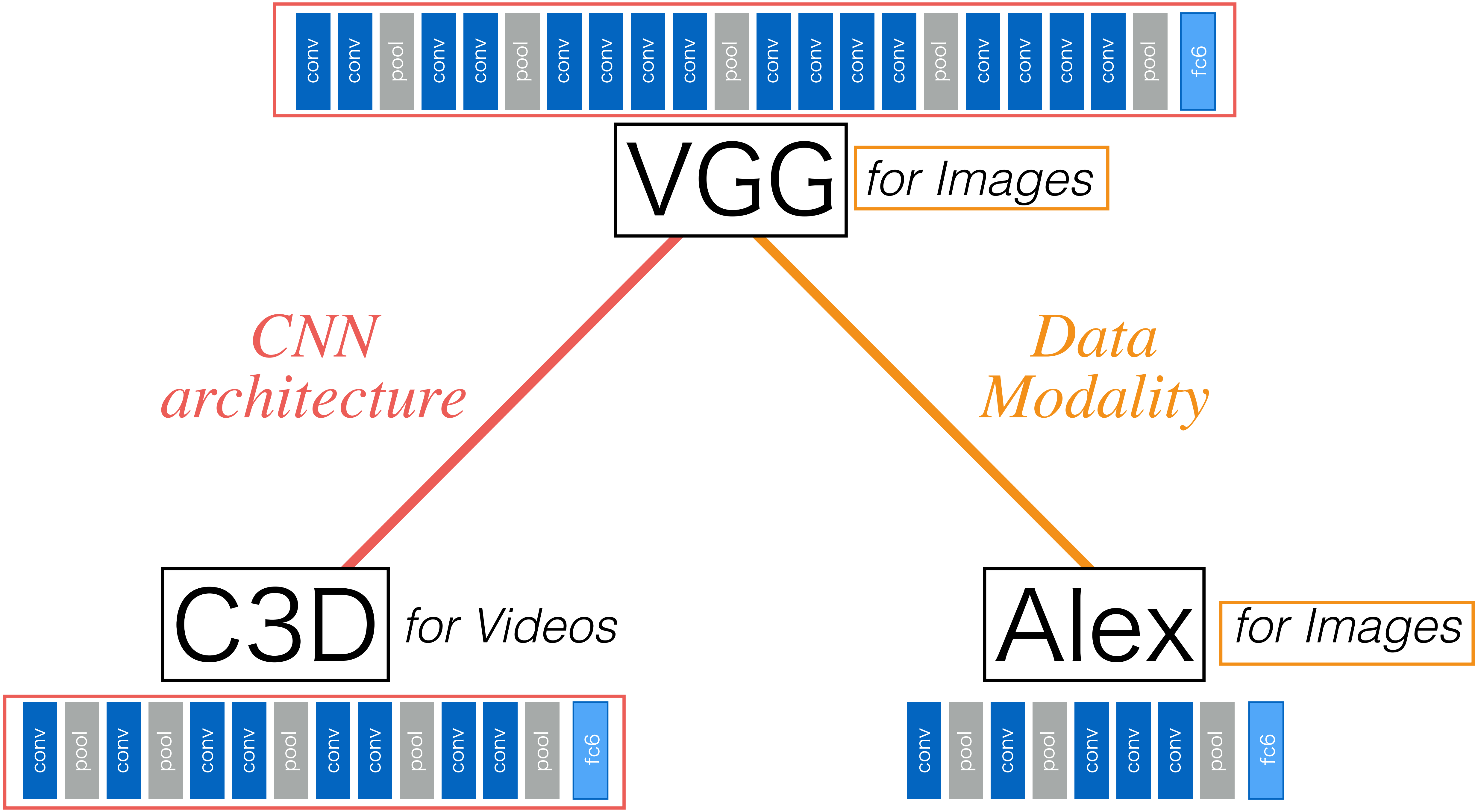
CNNモデル間で大体似通ったカーネル関数が得られた



(a) C3D (b) Alex (c) VGG

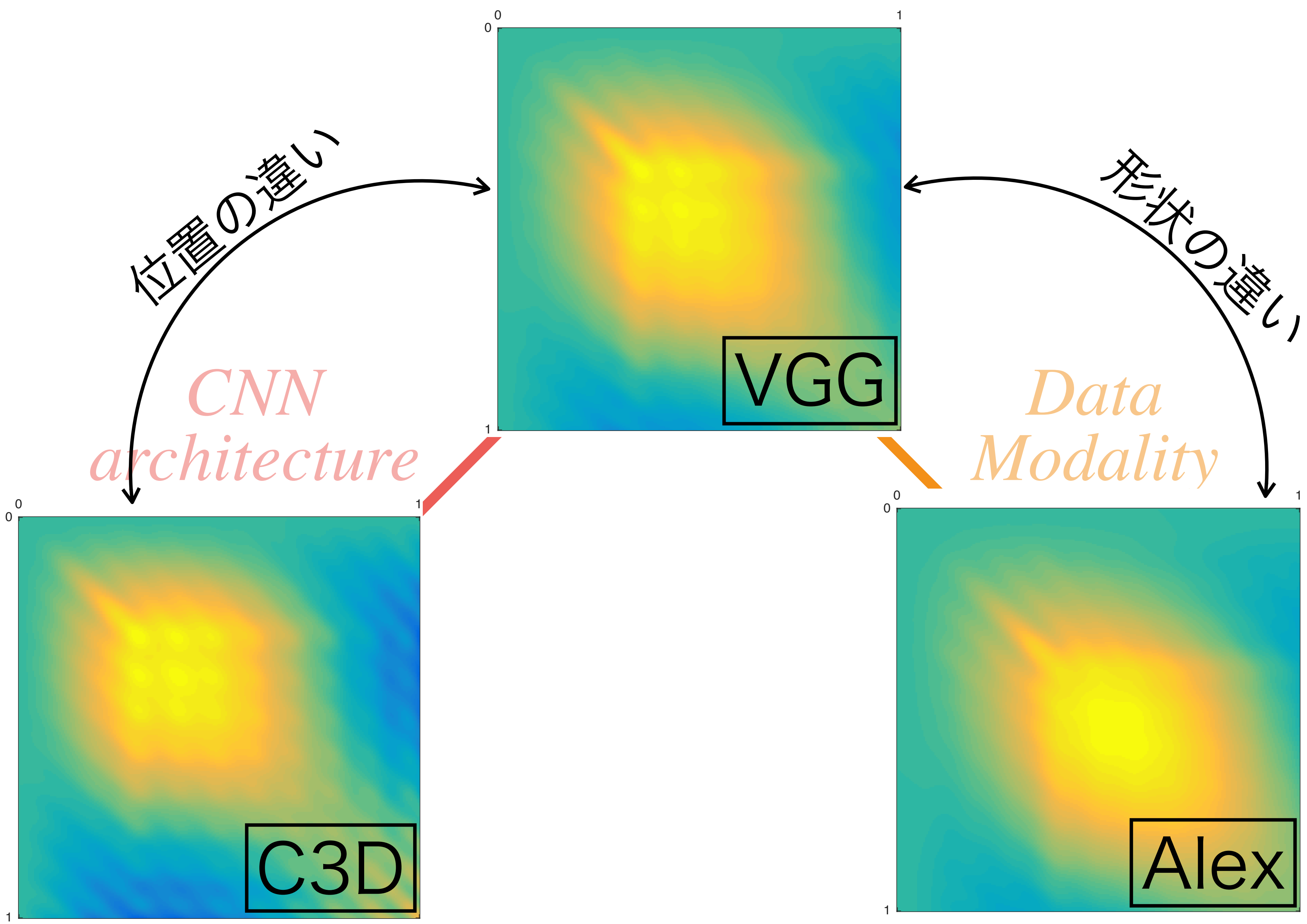
CNN特徴の関係性

CNN構造と学習データの特性を反映している



CNN特徴の関係性

CNN構造と学習データの特性を反映している



まとめ

人為的特徴量と学習的特徴量に対しての特徴変換

人為的特徴 = hand-crafted特徴：SIFT, HOGなど

学習的特徴 = CNN特徴 : AlexNet, VGGなど

人為的特徴量に対しては内在している**テンソル構造**に着目し、**学習的特徴量**に対しては**距離尺度の学習**を通して、それぞれの特徴変換手法を示した。

学習されたカーネル関数（の類似性）は、CNN特徴の特性解析に使えるかも…

ご清聴ありがとうございました