Машинное обучение (Machine Learning)

Деформирующие автокодеры и распутывание (Deforming autoencoders - DAE and Disentangling)

Уткин Л.В.

ПОЛИТЕХ
Санкт-Петербургский политехнический университет Петра Великого

Порождающие модели

1. Вариационные Автокодеры - Variational Autoencoders – VAE
2. Генеративные состязательные нейросети - Generative Adversarial Networks - GAN
3. Деформирующие Автокодеры - Deforming autoencoders - DAE

Классические автокодеры

- Могут обучаться для генерации компактных представлений
- Хорошо восстанавливают исходные данные
- **Но** фундаментальной проблемой автокодеров является то, что скрытое пространство, в котором они кодируют входные данные, может не быть непрерывным и не допускать гладкой интерполяции!

Вариационные Автокодеры (VAE)

- Могут решить эту проблему, так как их скрытое пространство является непрерывным и позволяет легко производить случайную выборку и интерполяцию
- **Но** управление глубокими нейронными сетями и, в особенности, глубокими автокодерами - сложная задача, ключевая особенность которой — строгий контроль процесса обучения

Деформирующий автокодер

- Это - породающая модель анализа изображений, которая выделяет признаки без дополнительных подсказок, предполагая создавать экземпляры объектов посредством деформации “шаблонного” объекта.
- Это означает, что вариативность объекта может быть разделена на уровни, связанные с пространственными трансформациями формы объекта.
- **Z. Shu et al. Deforming Autoencoders: Unsupervised Disentangling of Shape and Appearance // arXiv:1806.06503, Jun. 2018.**

Disentangling (распутывание)

О понятии Disentangling (распутывание)

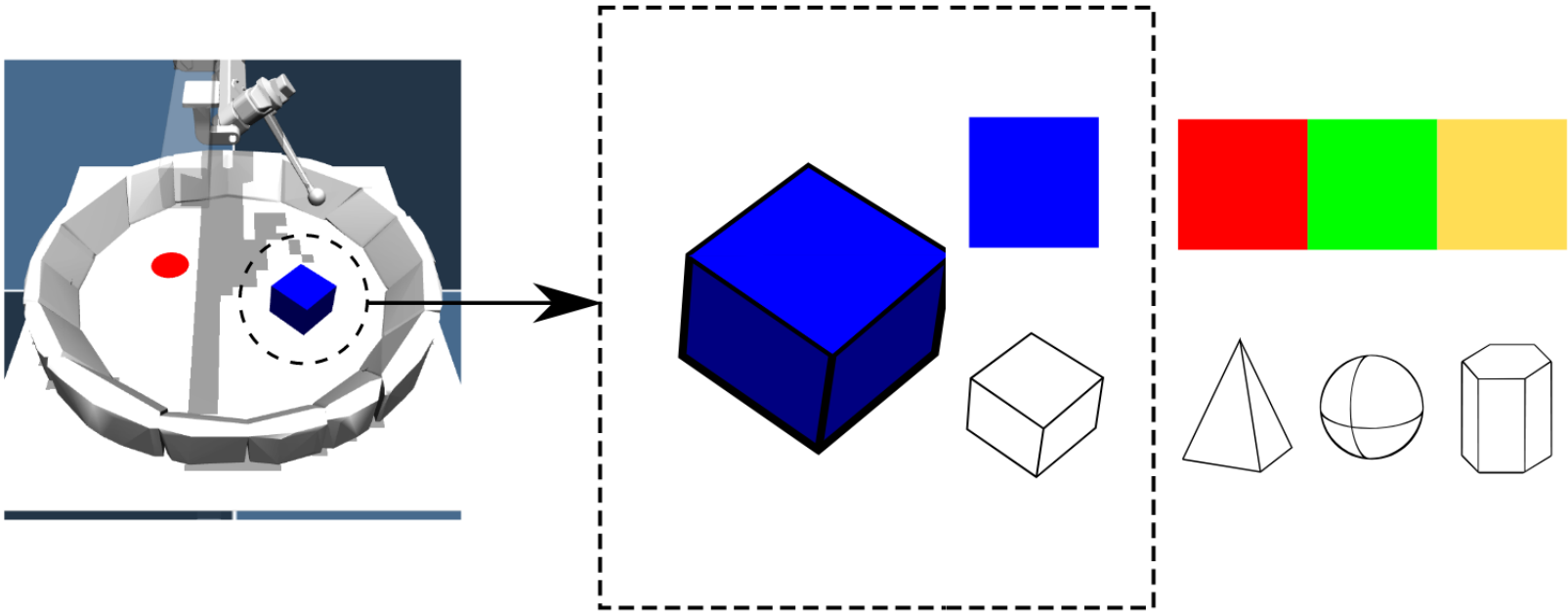
- Предположим, что следующие векторы являются соответственно представлениями для мяча: [1,0,0,0] и автомобиля: [0,1,0,0]
- В этом представлении один нейрон узнает значение мяча или автомобиля, не полагаясь на другие нейроны. Это распутанное представление.

Распутанные представления в интуитивном смысле означают, что скрытые факторы, которые изучает нейронная сеть, имеют семантическое значение.

Распутывание (1)

- Центральная мотивация распутанных представлений - идентификацию элементарных “строительных блоков” окружающего нас мира, которые неявно хранятся в данных.
- Эти “блоки” считаются инвариантными к изменениям, что делает их полезными для любой последующей задачи.
- Т.о. распутанное представление обычно предполагается как представление, которое отделяет или распутывает лежащую в основе структуру мира на непересекающиеся части его представления.

Распутывание (2)



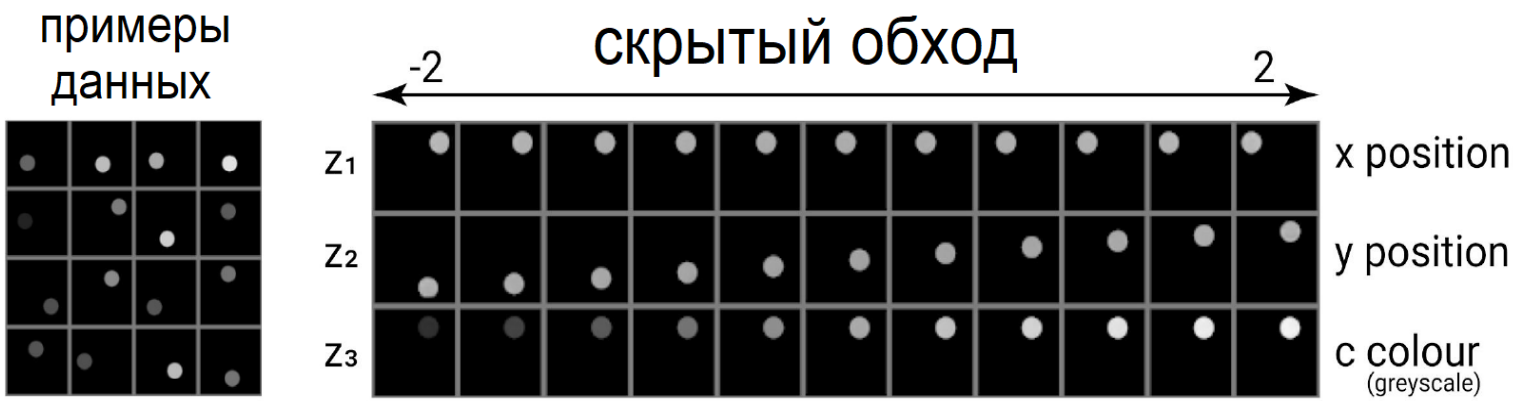
Слева - моделирование среды робота. Объект в среде представляет собой синий куб, поэтому распутанное представление может рассматривать непересекающиеся свойства цвета (синий) и формы (куб).

Распутывание (3)

- Например, это означает, что для трехмерной сцены с объектами скрытое представление должно отдельно кодировать размер, цвет, форму и положение.
- Распутывание было расплывчатой концепцией, которой интересовались многие, но не могли выразить ее явно.
- Что означает понятие “семантически значимые (meaningful) скрытые представления”.

Распутывание снова

- Визуально это то, что мы ожидаем: для точек в оттенках серого на 2D-плоскости мы хотим иметь x, y - позиции и цвет



Свойства распутывания - модульность

- **Модульность** измеряет, кодирует ли одна скрытая размерность не более одного фактора генерации данных.
- **Пример**: Когда изменение скрытого фактора z_i изменяет только один атрибут, например размер объекта, то он является модульным.
- **Контрпример**. Если изменение z_i меняет и цвет и размер, то он не модульный в этом смысле.

Свойства распутывания - компактность или полнота

- **Компактность/полнота** измеряет, кодируется ли каждый фактор генерации данных одной скрытой размерностью.
- **Пример**: Полнота требует, чтобы атрибут изменялся только при изменении конкретного z_i. Для всех $z_{j \neq i}$ атрибут (например, цвет) должен оставаться постоянным.
- **Контрпример**. Полнота - обратное модульности. Модульность по-прежнему выполняется, если оба z_i и z_j кодируют цвет, но такое представление не является компактным.

Свойства распутывания - информативность (1)

- **Информативность** измеряет, могут ли значения всех факторов, генерирующих данные, быть декодированы линейным преобразованием. Распутанное представление должно охватывать все скрытые факторы (условие 1), и эта информация должна быть линейно декодируемой (условие 2).
- **Пример**: В 3D-сцене объекта с определенной формой, размером, положением и ориентацией все факторы соответствуют скрытым факторам, поэтому можно извлечь всю информацию, применив линейное преобразование, т.е. $z_{true} = Az_{learned}$. Т.е. может случиться так, что один $z_{learned,i}$ изменяет несколько факторов, но можно найти такую матрицу A, что получим факторы, для которых выполняется модульность.

Свойства распутывания - информативность (2)

- **Информативность** измеряет, могут ли значения всех факторов, генерирующих данные, быть декодированы линейным преобразованием. Распутанное представление должно охватывать все скрытые факторы (условие 1), и эта информация должна быть линейно декодируемой (условие 2).
- **Контрпример**. Условие 1 нарушается, если, например, цвет не закодирован в латентном состоянии; а условие 2 не выполняется, если не существует такой матрицы A, для которой выполняется $z_{true} = Az_{learned}$ (например, существует нелинейное отображение в z_{true}.

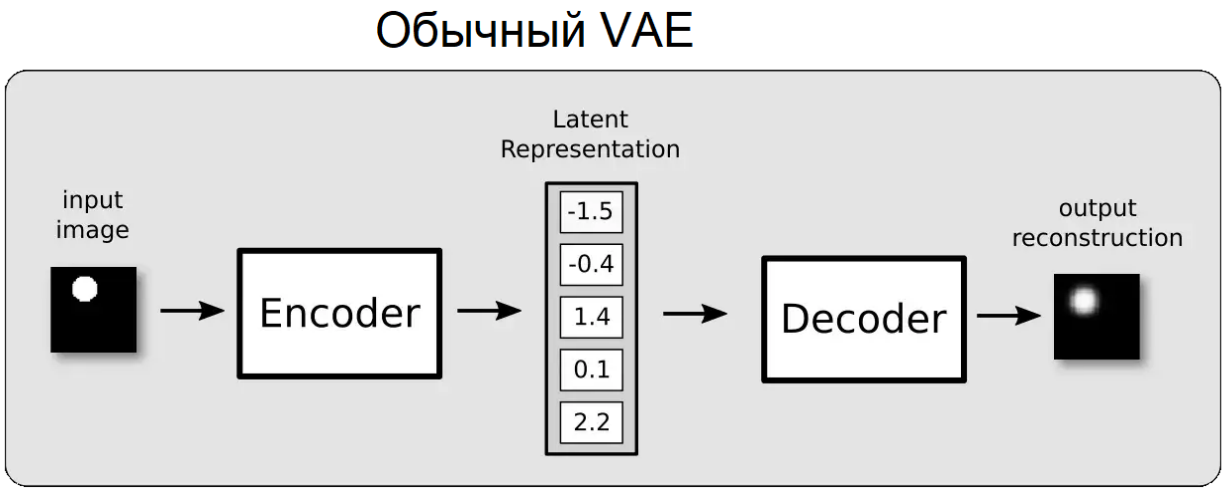
Распутывание с VAE (1)

- β-VAE пытается решить сложную проблему - найти независимые порождающие факторы набора данных без учителя, чтобы получить факторизованную генеративную модель. Это - супер-проблема.
- Как β-VAE делает это? Гиперпараметр β управляет информационным латентным слоем в VAE, “поощряя” распутанные скрытые представления. Один из способов визуализировать это - использовать скрытые обходы (latent traversals).
- Скрытый обход - начинаем со случайно выбранного примера данных и пропускаем его через кодировщик VAE, получая скрытое представление $z \in \mathbb{R}^N$ примера.

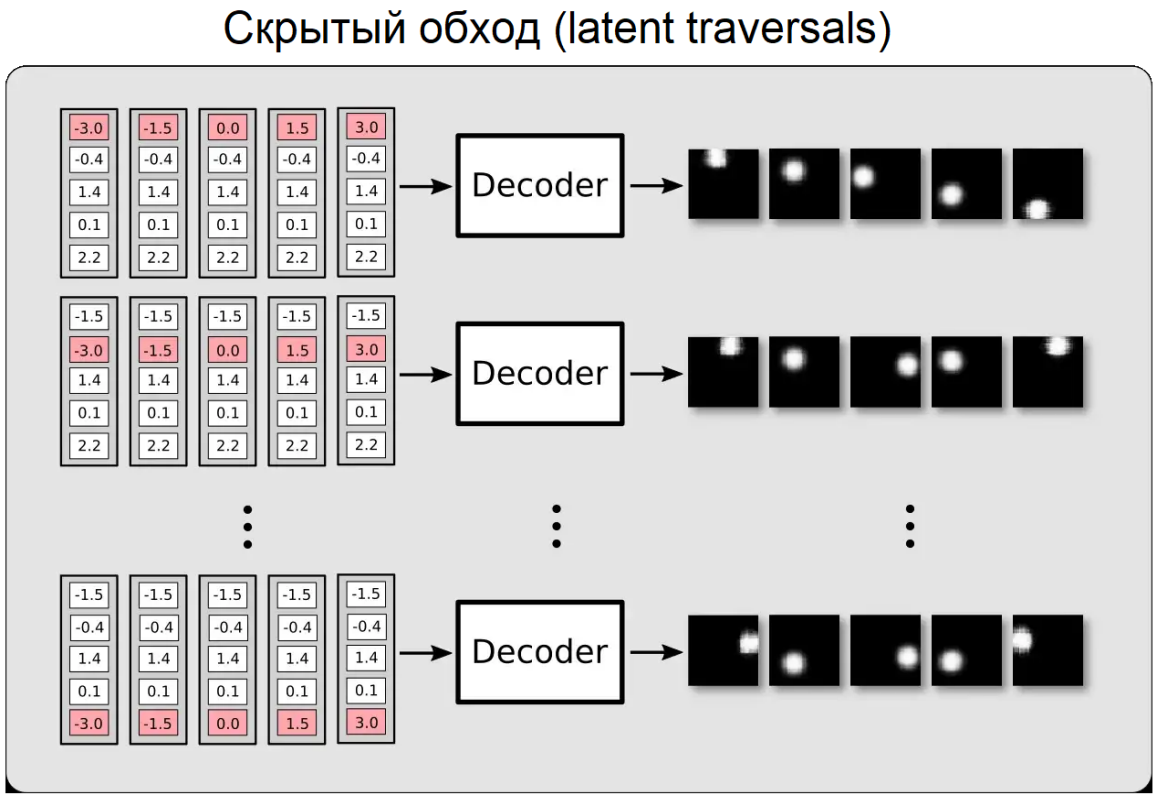
Распутывание с VAE (2)

- Если скорректиравть один элемент вектора z, сохранив фиксированными другие $N - 1$ элементов, то можно создать множество скрытых вариаций, которые затем декодируются. Процедура повторяется для всех элементов z.
- Для воссоздания фигуры будем использовать простой набор данных, состоящий из изображений белой точки на черном фоне. Точка всегда одного размера, и единственное, что меняется, — это ее местоположение. Таким образом, есть два основных генерирующих фактора: координаты x и y.

Просто VAE


Обычный VAE
Latent
Representation
input
image
Encoder
-1.5
-0.4
1.4
0.1
2.2
Decoder
output
reconstruction

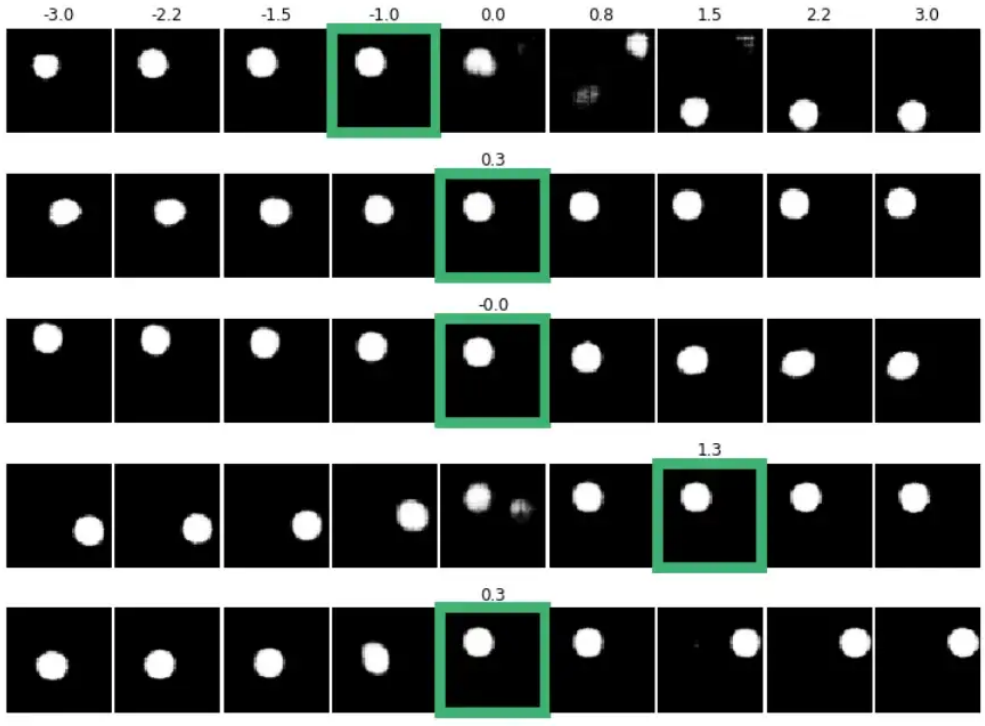
Распутывающий VAE



Распутывающий VAE с малым параметром (1)

- Скрытый обход для β-VAE, где значение β слишком малое.
- В этом случае латентный слой слишком широкий, и сеть не может создать эффективное скрытое представление.
- Сеть распределяет скрытое представление по четырем различным измерениям. Напомним, что существует только два порождающих фактора: x и y, поэтому двух скрытых элемента было бы достаточно.
- Более того, когда корректируется один скрытый элемент, выходные данные не всегда являются достоверными реконструкциями, некоторые из них даже показывают две точки.

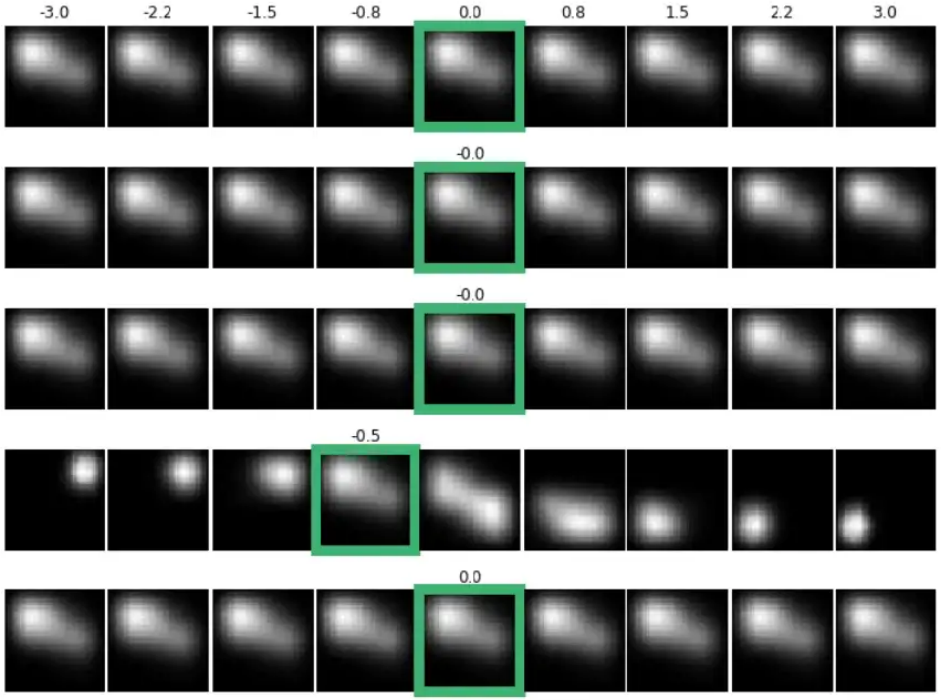
Распутывающий VAE с малым параметром (2)



Когда латентный слой большой, сеть свободно кодирует запутанное и неэкономное скрытое представление. Зеленые границы обозначают реконструкции с неизмененными скрытыми элементами.

Распутывающий VAE с большим параметром

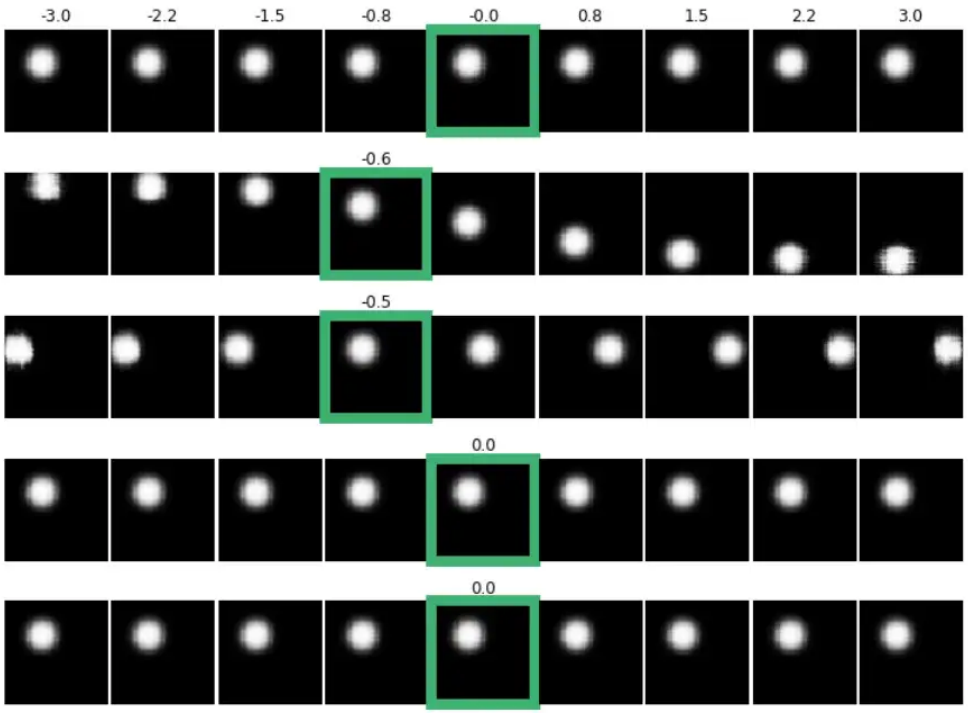
- Увеличим β



Значение β слишком велико. Латентный слой слишком мал, что вынуждает сеть пытаться закодировать два порождающих фактора в одном скрытом элементе, сеть не может делать достаточно хорошие реконструкции

Распутывающий VAE с “хорошим” параметром

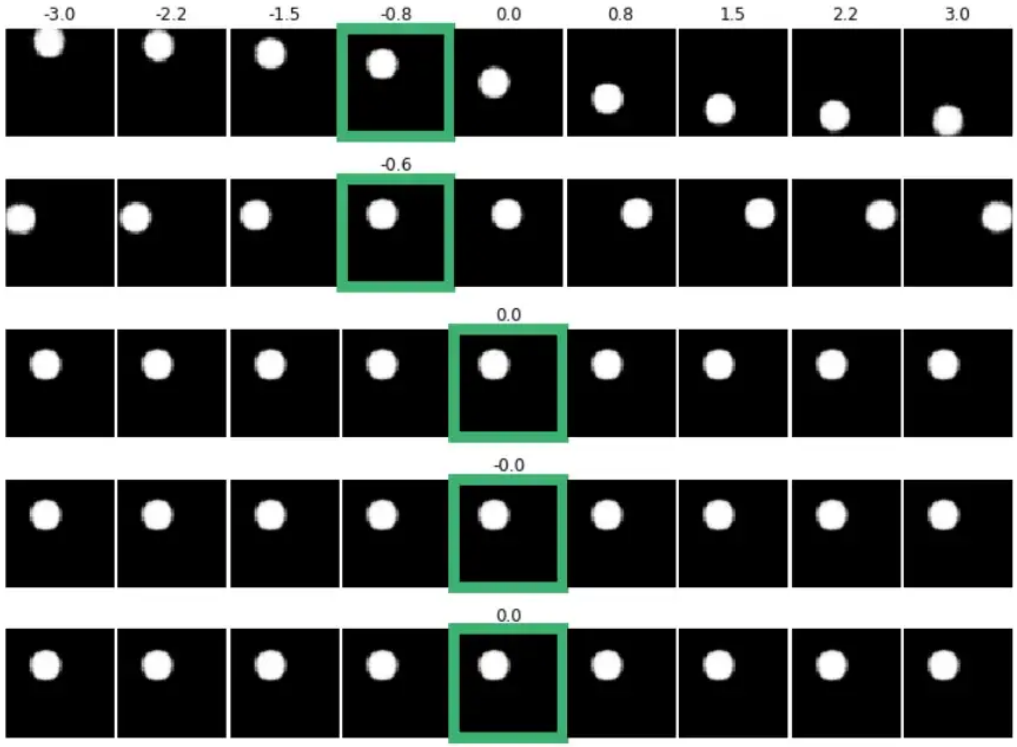
- Имеется теперь скрытое пространство, в котором только два из пяти доступных измерений на самом деле что-то кодируют, и более того, они ортогональны.



Распутывающий VAE с “хорошим” параметром

- Пострадали в отношении качества реконструкции (по сравнению с малым β), но, это можно исправить.
- Нужно начинать с высокого значения β и уменьшать его во время обучения.
- Скрытый обход сети, обученной со значением β, скорректированным во время обучения. Распутанное представление кодируется с сохранением высокого качества реконструкции.

Распутывающий VAE

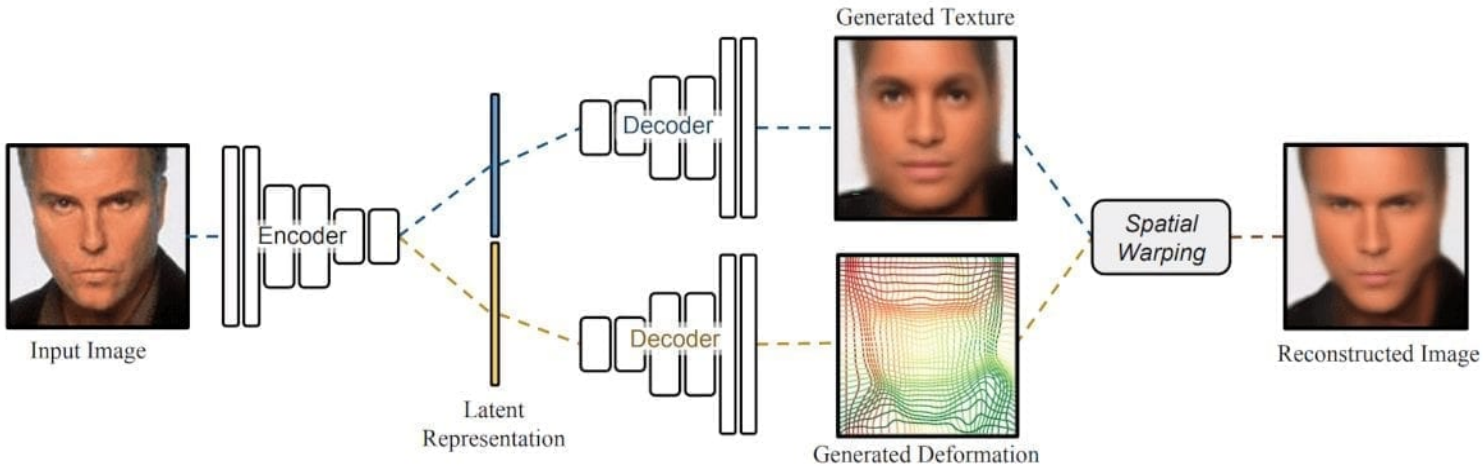


Скрытый обход сети, обученной со значением β, скорректированным во время обучения. Распутанное представление кодируется с сохранением высокого качества реконструкции

Снова деформирующий автокодер

- Деформирующий автокодер способен определять форму и внешний вид объекта как степени вариативности в изученном малоразмерном скрытом пространстве
- Архитектура состоит из
  - кодера, который кодирует входное изображение в два скрытых вектора (один – для формы, другой – для вида)
  - двух декодеров, принимающих векторы в качестве входных данных и выдающих сгенерированную текстуру и деформации
- Независимые декодеры необходимы для получения функций внешнего вида и деформации

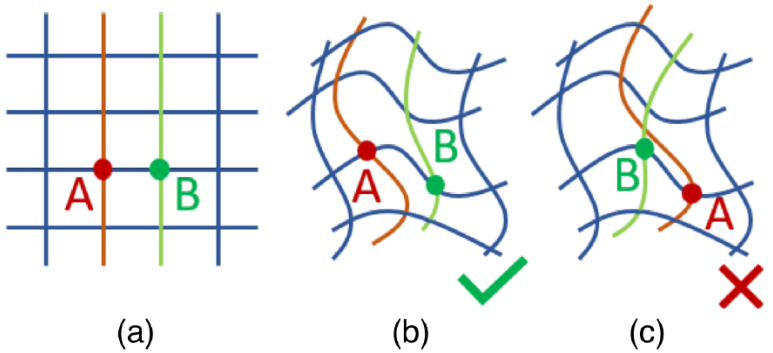
Структура (1)


Generated Texture
Decoder
Encoder
Spatial Warping
Input Image
Latent Representation
Decoder
Generated Deformation
Reconstructed Image

Структура (2)

- Сгенерировнная пространственная информация используется для деформации текстуры к наблюдаемым координатам изображения
- DAE может восстановить входное изображение и в то же время определеить форму и вид объекта как различные особенности
- Вся нейросеть тренируется без помощника на основе лишь простых потерь восстановления изображения

Разрешенные деформации

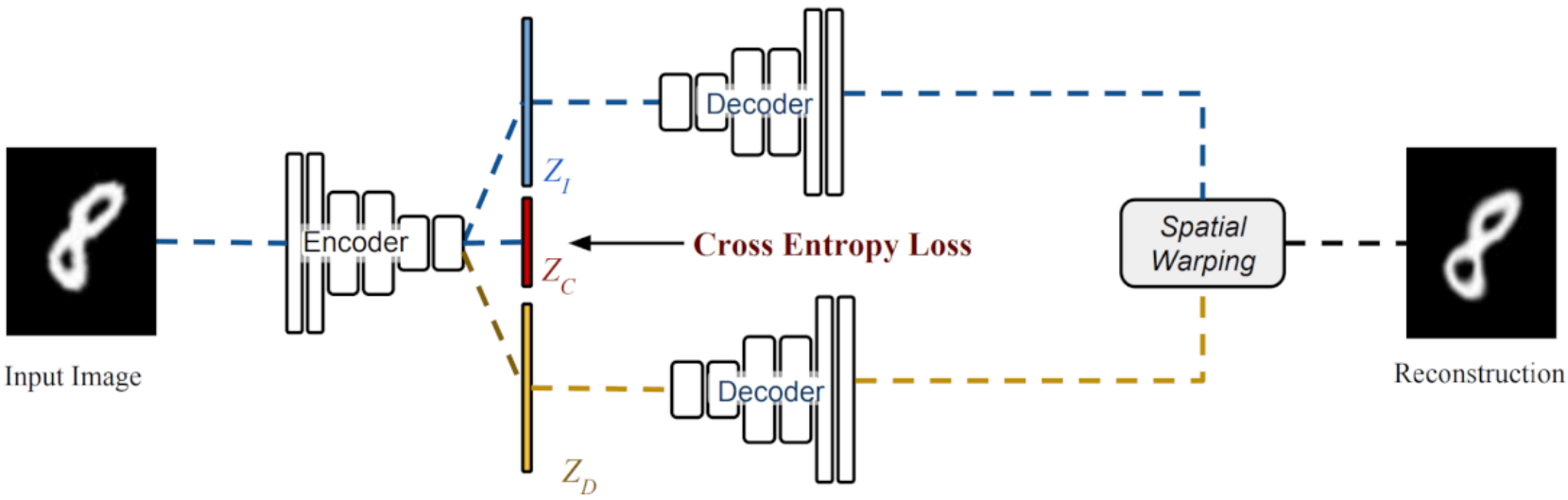


- Допускаются только локально согласованная деформации (b)
- Изменение относительных позиций пикселей (c) не допускается
- Для этого позволяем декодеру деформации предсказывать горизонтальные и вертикальные приращения деформации ($\nabla_x W$ и $\nabla_y W$)

DAE с заданной классификацией

- **DAE с заданной классификацией** учатся восстанавливать изображение и одновременно определяют форму и вид факторов вариативности, соответствующие определенному классу
- Для реализации, вводят класифицирующую нейросеть, вход которой - третий скрытый вектор, используемый для кодирования класса. Это позволяет изучать смешанную модель, обусл-ную классом вх. изобр.
- Это улучшает эффективность и стабильность обучения, так как нейросеть учится разделять типы пространственной деформации, различные для каждого класса

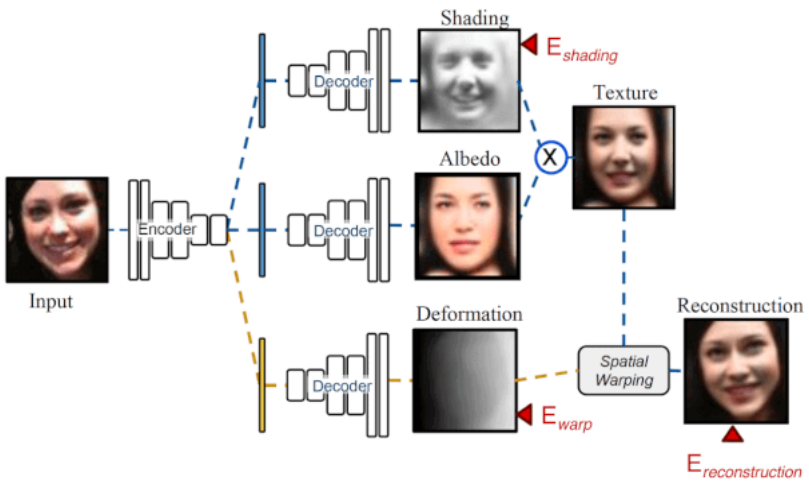
DAE с заданной классификацией

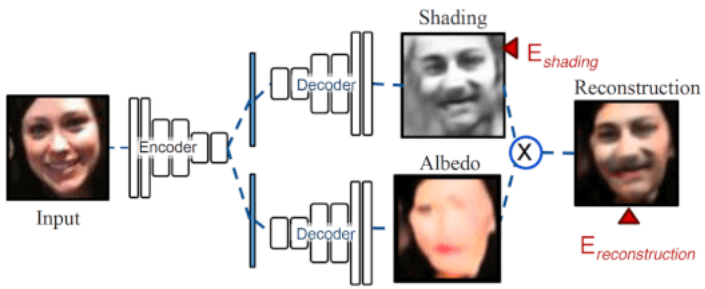


Встроенный DAE

Для вычисления альбедо и теней на портретных изображениях

Альбедо - коэффициент диффузного отражения, то есть отношение светового потока, рассеянного плоским элементом поверхности во всех направлениях, к потоку, падающему на этот элемент





Обучение (1)

$$E_{\text{DAE}} = E_{\text{Reconstruction}} + E_{\text{Warp}}$$

- Reconstruction loss and warping loss

$$E_{\text{Reconstruction}} = \|\text{Output} - \text{Input}\|^2,$$
$$E_{\text{Warp}} = E_{\text{Smooth}} + E_{\text{BiasReduce}}$$

- E_{Smooth} штрафует быстро меняющиеся деформации, закодированные локальным полем деформации, измеряется как общая норма вариации горизонтальных и вертикальных деформаций
- $E_{\text{BiasReduce}}$ направлены на устранение любого систематического искажения, вносимого процессом подбора

Обучение (2)

$$E_{\text{Smooth}} = \lambda_1 \left(\|\nabla W_x(x,y)\|_1 + \|\nabla W_y(x,y)\|_1 \right)$$

- $\nabla W_x(x,y)$ и $\nabla W_y(x,y)$ смещения по x и y

$$E_{\text{BiasReduce}} = \lambda_2 \left(\|S_A - S_0\|^2 + \|W - W_0\|^2 \right)$$

- тень S и альбедо A

MNIST

- DAE способен успешно выявлять форму и внешний вид объектов во время обучения
- DAE с заданной классификацией дают наилучшие результаты как при восстановлении, так и при изучении внешнего вида объектов

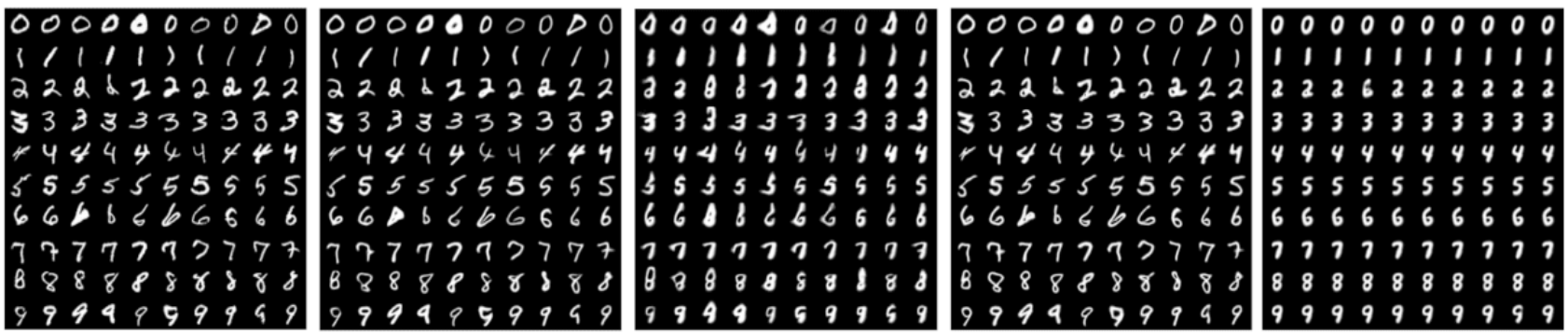
(a) input (b) reconstruction (c) decoded appearance (d) decoded deformation

DAE в различных режимах (1)

- Наложение изображений без учителя.
- Изучение семантически важных множеств для формы и внешнего вида объектов.
- Внутренняя декомпозиция без учителя.
- Детектирование локализации без учителя.

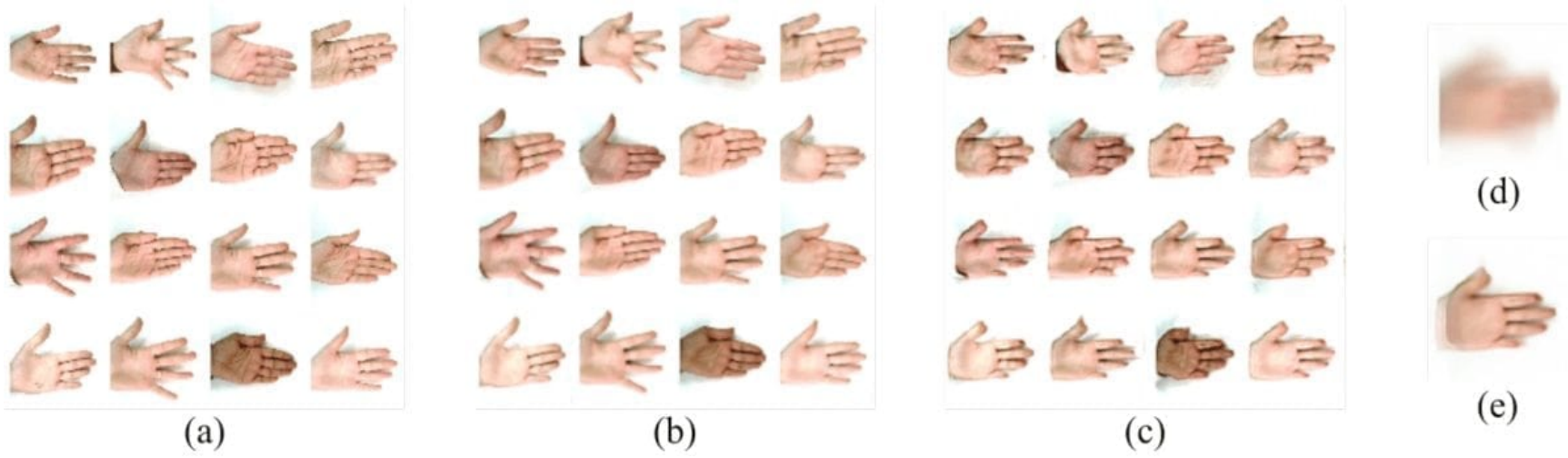
DAE в различных режимах (2)

Восстановления MNIST с заданной классификацией



(a) input (b) Reconstruction w/o classification (c) Appearance w/o classification (d) Reconstruction with classification (e) Appearance with classification

Наложение изоб-ний ладоней без учителя



(a) – Входные изображения (b) – восстановленные изображения (c) – изображения текстур, деформированные с использованием среднего декодированной деформации (d) – среднее входное изображение (e) – средняя текстура

Интерполяция освещения с помощью встроенного DAE

И так...

- DAE - специфическая архитектура, способная выявлять определ҆нные факторы вариативности – в данном случае это форма и внешний вид объектов.
- Результаты работы DAE показывают, что она способна успешно выявлять факторы вариативности посредством применения архитектуры автокодеров

Вопросы

?