CV：hand pose estimation

1. 目前进展

1.1 相关资料

1) HANDS CVPR 2016
2) HANDS 2015 Dataset
3) CVPR 2016
4) Hand 3D Pose Estimation (Computer Vision for Augmented Reality Lab)
5) CVPR2016 Tutorial: 3D Deep Learning with Marvin
6) MSRA Hand Dataset
7) NYU Depth Dataset V2
8) Hand Gesture Datasets RGB-D Dataset from Multimedia Technology and Telecommunications Laboratory
9) HandNet RGB-D Dataset
10) LibHand
11) Dexter 1 Dataset
12) Vehicles and Applications (VIVA) Challenge
13) Hand Pose Estimation & Hand Gesture Recognition（Papers Reading List in CVPR 2016）
14) HAND DATASETS AND METHODS
15)

1.2 流程

基本流程如下：
1) Detect and Segment Hand
2) Estimate Pose
3) Validate or Refine

1.3 困难

手势识别困难之处：
1）手分辨率低（Low-res hand）
2）背景杂乱（Clutter background）

3）手与其它对象交互（Object/surface interaction）
4）手被遮挡（Occlusions/Self-occlusions）
5）不同手势相似（Self-similarity）
6）多自由度（many DoF(Degree of Freedom)）
7）多视角（Multiple viewpoints）
8）不同的形状和尺寸

1.4 组件

1）Training sets
2）Testing sets
3）Models

1.5 生成方法与判别方法

数据驱动方法（Data-Driven）更有优势，因为它不需要复杂的模型校准；且即使初始化很差，其结果照样健壮（即对初始化的依赖性不高）。

- holistic (coarse to fine)
- 手势识别：Hand Gesture Recognition
- 动作识别：Action Recognition
- 手势估计方法：生成方法和判别方法

1.5.1 生成方法（Generative Methods）

生成方法（基于模型）（Generative mthods: model-based）
- 步骤：首先，创建大量的手势；然后，选择一个最匹配当前深度图像的手势
- 目标函数（objective function）：基于输入深度图与手模型近似深度图的相似性，然后对此目标函数进行优化，以找到最接近的手模型。
- 缺点：
（1）优化(找最匹配的)计算量大
（2）其精确性高度依赖人工创建的相似性函数（similarity function）
（3）如果前面的估计不准确，易于出现错误累积
（4）为减轻普遍存大的模型漂移（model drift），近来采用“优化+重新初始化”范式

1.5.2 判别方法（Discriminative Approaches）

判别方法（基于外貌）（Discriminative approaches：appearane based）
- 学习从深度图像到手势配置的映射（手势配置 = mapping(深度图像)）
- 手深度图低分辨率、自我遮挡、快速移动会产生大量错误

- 基于局部回归（local regression）的方法：可以提高对遮挡的鲁棒性，但是易产生帧间抖动

1.6 手势姿态估计方法

1.6.1 方法分类

1）追踪与检测（Trackers versus Detectors）：
检测：基于单帧的方法，每帧都会重新初始化它自己
追踪：基于多帧的方法，不能从错误中立即恢复

2）数据驱动与模型驱动（Data-driven vensus Model-driven）：

模型驱动：拿着模型找与之匹配的数据（已经知道本质规律，来对数据进行判断）；优化一个几何模型以拟合观察到的数据；其目标函数容易出现局部最优；它在追踪领域取得了很大的成功，它的初始化限制了搜索空间

Model-driven approach：This parametric modelling approach consists of searching the most appropriate model among basic building models contained in a models library.

数据驱动：拿着数据总结模型（根据已知数据寻求本质规律）；对于单个图像检测，各种快速的分类算法可以实时地实现；这些分类器由几何模型合成的数据进行训练，可以看作是模型的近似拟合

Data-driven approach, also called non-parametric modelling approach：This technique attempts to model a primitive or a complex building by using series of more or less complex operations. It allows the generation of a model without belonging to a specific library.

Model-Driven	**Data-Driven**
User interacts primarily with a (mathematical) model and its results	User interacts primarily with the data
Helps to solve well-defined and structured problem (what-if-analysis)	Helps to solve mainly unstructured problems
Contains in general various and complex models	Contains in general simple models
Large amounts of data are not necessary	Large amounts of data are crucial
Helps to understand the impact of decisions	Helps to prepare decisions by showing developments in the past and by identifying relations or patterns

3）多阶段管道（Multi-stage pipelines）：
一般分为pre-processing stage（Segmentation）和post-processing stage

1.6.2 方法汇总

Method	Approach	Model-drv.	Data-drv.	D
Simulate [23]	Tracker (simulation)	Yes	No	Init
NiTE2 [32]	Tracker (pose search)	No	Yes	Init
Particle Swarm Opt. (PSO) [28]	Tracker (PSO)	Yes	No	Init
Hough Forest [51]	Decision forest	Yes	Yes	Dec
Random Decision Forest (RDF) [20]	Decision forest	No	Yes	
Latent Regression Forest (LRF) [45]	Decision forest	No	Yes	
DeepJoint [47]	Deep network	Yes	Yes	Dec
DeepPrior [26]	Deep network	No	Yes	Scan
DeepSegment [12]	Deep network	No	Yes	Scan
Intel PXC [18]	Morphology (convex detection)	No	No	Heuri
Cascades [35]	Hierarchical cascades	No	Yes	Scan
EPM [53]	Deformable part model	No	Yes	Scan
Volumetric Exemplars	Nearest neighbor (NN)	No	Yes	Scan

Table 4 Summary of methods: We broadly categorize the pose estimation systems `approach`: decision forests, deep models, trackers, or others. Though we focus on single-fram by providing them manual initialization. `Model-driven` methods make use of articulated ge `data-driven` models are trained beforehand on a training set. Many systems begin by `detect` or a scanning window/volume search. Finally, we made use of public source code when avail ourselves, verifying our implementation's accuracy on published benchmarks. 'Published' in results were used for evaluation, while 'public' indicates that source code was available, allo additional testsets. We report the fastest speeds (in FPS), either reported or our implement

1.6.3 架构

1）Decision forests

2）Part Model

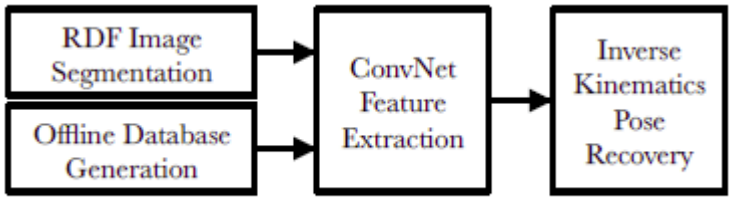
3）Deep Models

-Deep-Joint：分为三阶段管理

(1) 使用Decision forest检测手

(2) 使用深度网络回归关节位置

(3) 使用IK优化关节预测



Pose Recovery Pipeline Overview

-Deep-Prior：

(1) 基于类似的深度网络

(2) 依靠网络本身学习到的“空间先验知识(Pose Prior)”来优化关节预测，而不是使用IK

(3) 使用Overlapping Regions来优化3D关节位置，使其精度更高；小的Region提供精确度，大的Region提供环境信息

-Deep-Segment：

(1) 采用像素标签方法，为每一个像素预测其属于的关节标签

(2) 通过聚类方法产生关节位置

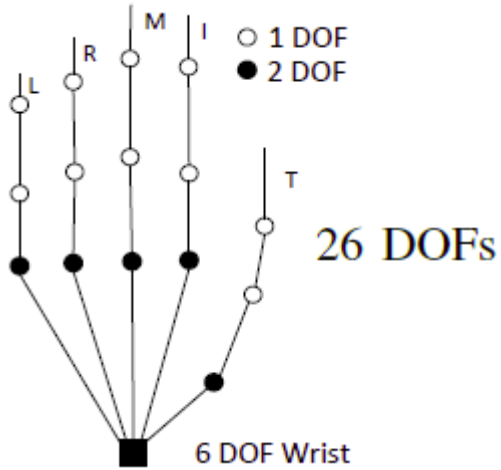
(3) 此方法让人想起了Kinect像素级部件分类方法，但是用deep network代替了decision forest

1.7 关节描述及自由度(DOF)

1）关节位置（joint position）：需要考虑全局方向（即摄像机视角），局限于视角

2）关节角度（joint angle）：不需要考虑全局方向（即摄像机视角），与视角无关

3）自由度(DOF)



1.8 训练数据及其生成方法

1）真实数据+手动标注（Real data + manual annotation）：ICL（标记了16个关节）
不同的人标记，差异高达20%

2）真实数据+自动标注（Real data + automatic annotation）：NYU（标记了36个关节）
可以采用被动的运动捕获系统来获取训练数据

3）准合成数据（Quasi-synthetic data）：
对真实数据及其标注进行几何变换（如旋转等），可以产生大量的合成数据

4）合成数据（Synthetic data）：UCI-EGO

数据由计算机图像系统渲染生成，精确的标注可以与渲染时一起产生，所以完全避免了标注的问题
可以通过开源的libhand(依赖: Boost, OpenCV, OGRE)模型来生成

Dataset	Generation	Viewpoint	Views	Size	Subj.
ICL [45]	Real + manual annot.	3rd Pers.	1	331,000	10
NYU [47]	Real + auto annot.	3rd Pers.	3	72,757	1
UCI-EGO [35]	Synthetic	Egocentric	1	10,000	1
libhand [50]	Synthetic	Generic	1	25,000,000	1

Table 3 Training data sets: We broadly categorize training datasets by the method used to **generate** the data and annotations: real data + manual annotations, real data + automatic annotations, or synthetic data (and automatic annotations). Most existing datasets are **viewpoint**-specific (tuned for 3rd-person or egocentric recognition) and limited in **size** to tens of thousands of examples. NYU is unique in that it is a multi**view** dataset collected with multiple cameras, while ICL contains shape variation due to multiple (10) **subjects**. To explore the effect of training data, we use the public libhand animation package to generate a massive training set of 25 million examples.

1.8 测试数据（都是真实数据）

Dataset	Chal.	Scn.	Annot.	Frms.	Sub.	Cam.	Dist. (mm)
ASTAR [51]	A	1	435	435	10	ToF	270-580
Dexter 1 [42]	A	1	3,157	3,157	1	Both	100-989
MSRA [33]	A	1	2,400	2,400	6	ToF	339-422
ICL [45]	A	1	1,599	1,599	1	Struct	200-380
FORTH [28]	AV	1	0	7,148	5	Struct	200-1110
NYU [47]	AV	1	8,252	8,252	2	Struct	510-1070
KTH [30]	AVC	1	0	46,000	9	Struct	NA
UCI-EGO [35]	AVC	4	364	3,640	2	ToF	200-390
Ours	AVC	10+	23,640	23,640	10	Both	200-1950

Challenges (Chal.): A-Articulation V- Viewpoint C-Clutter

Testing data sets: We group existing benchmark testsets into 3 groups based on the overall **challenges** addressed - articulation, viewpoint, and/or background clutter. We also tabulate the number of captured **scenes**, number of **annotated** versus **total frames**, number of **subjects**, **camera** type (structured light vs time-of-flight), and **distance** of the hand to camera. We introduce a new dataset (**Ours**) that contains a significantly larger range of hand depths (up to 2m), more scenes (10+), more annotated frames (24K), and more subjects (10) than prior work.

2. 生成模型（Model Based Methods）

主要问题：初始化复杂、容易陷于局部最优

【1.1】(2014) Realtime and Robust Hand Tracking from Depth.

Chen Qian; Xiao Sun, Yichen Wei, Xiaoou Tang, Jian Sun

- 只使用CPU可达到25FPS，误差小于10mm

- 使用48个球简单定义手模型，并且定义了一个快速的cost函数；

- 定义了一个自由度（DOF）为26 的手的模型，其中的6个自由度代表全局的手的模型（整体）；每一个手指还有4个自由度（总共20个）。同时还加上了手的运动学的限制。

- 使用基于梯度的随机优化方法，使其快速收敛并获取好的精确度；

1）基于梯度的优化方法。但是明显的缺点是：很容易被陷在局部最优里面，同时对于非刚体的处理效果也不是很好。

2）PSO: 是一种从参数空间搜索最优化参数的方法。在演化的过程中，整个点云的最好位置以及以及每一个点的局部最好的位置都被记录下来。在每一个的演化的过程中，粒子的速度都以它前一个速度的某种运算来更新。这样得到的结果的好处是：能够更好的搜索参数空间，避免那些很差的局部最优值。但是缺点是：速度比较慢。

- 建议了一个新的手指检测和手初始化方法

- Video

- MSRA Hand Dataset :Benchmark

3. 判别模型（Data-Driven，Learning Based Methods）

【1】(2014.6) Real-Time Continuous Pose Recovery of Human Hands Using Convolutional Networks. Jonathan Tompson, Murphy Stein, Yann LeCun and Ken Perlin.

- Deep-Joint

- 使用CNN提取特征，并为关节位置生成小的热图(heatmaps)，然后从特征和小的热图中使用IK(反向动力学)推断出手的姿势。

- 此方法只能预测关节的2D位置，然后使用深度图算出第三个坐标，这对于隐藏关节是有问题的。此外，其精确度受限于heatmap分辨率；因为CNN必须在每个像素位置进行评估，所以创建热图计算量很大。

- 40FPS (without CPU，见"2015-Cascaded Hand Pose Regression")

- Code

- NYU Hand Pose Dataset : Benchmark

1) 使用Primesense Carmine 1.09(结构光)抓取 RGB-D数据(每一帧的关节位置通过3个Kinect获取)

2) 72K训练样本(1人)，8K测试帧(2人)

3) Ground Truth 标记包含36个关节，本文只使用了14个关节

【2】(2014.4) Latent Regression Forest: Structured Estimation of 3D Articulated Hand Posture.

Danhang Tang， Hyung Jin Chang， Alykhan Tejani， Tae-Kyun Kim

- 粗略估计一个包含手的3D边框

- 主要贡献：

1）学习手的拓扑结构（以非监督、数据驱动的方式），它由Latent Tree Model表示

2）一个新的基于森林的判别框架（LRF: Latent Regression Forest），此框架在图像中进行结构化搜索（coarse-to-fine），同时框架每个阶段嵌入一个错误回归器以避免错误累积

3）生成一个多视角的手势Dataset（180K 3D标记深度图，从10不同的测试者采集）

- ICVL Hand Posture Dataset：Benchmark

1) 使用Intel Creative TOF深度摄像头， 22K训练样本
2) 每行对应一个图像(包含16x3个数字，表示16个关节的位置（x,y,z)，且是关节中央的位置)
3) (x,y)以像素为单位，z以mm为单位
4) 16个关节的顺序：Palm, Thumb root, Thumb mid, Thumb tip, Index root, Index mid, Index tip, Middle root, Middle mid, Middle tip, Ring root, Ring mid, Ring tip, Pinky root, Pinky mid, Pinky tip

5) 不精确的标记

- Video Awesome Random Forest Danhang Tang

- 62.5FPS (without CPU，见"2015-Cascaded Hand Pose Regression")

【3】(2015.2) Hands Deep in Deep Learning for Hand Pose Estimation.

Markus Oberweger, Paul Wohlhart, Vincent Lepetit

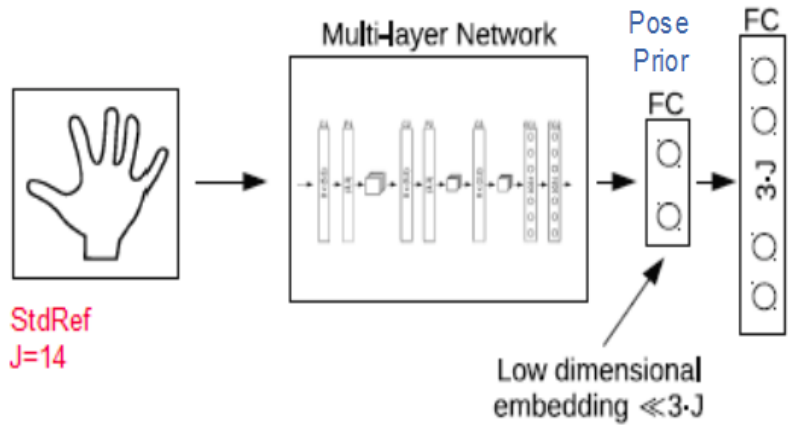
- Deep-Prior

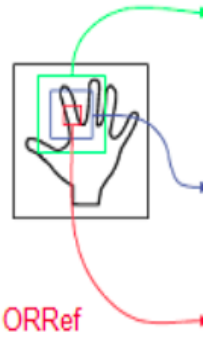
- 使用CNN网络直接深度图中手关节的位置。本文的特点是速度很快并且精度可以通过refinement提高。作者主要的贡献是两个部分:
1）设计一个加入了prior的网络输出手的关节点
2）基于上述关节点预测，对每一个关节点用一个refinement网络来进行更精确的关节点输出。甚至可以用迭代的方式多次refine关节点位置

- Code Python Code based on Caffe

- 引用【1】、【2】

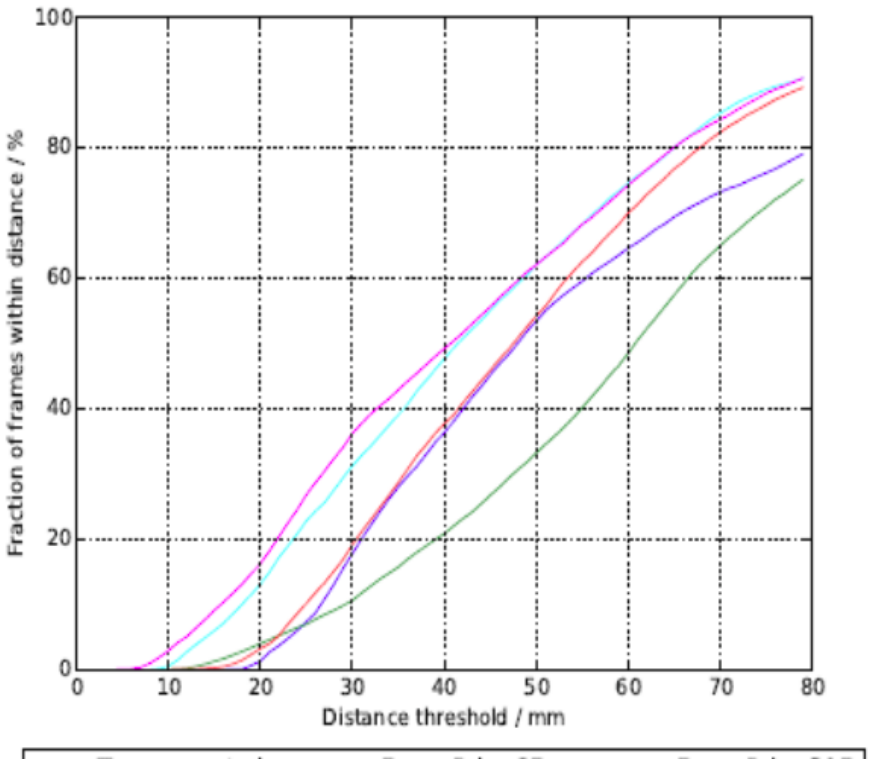
2015 - Oberweger (Deep-Prior)

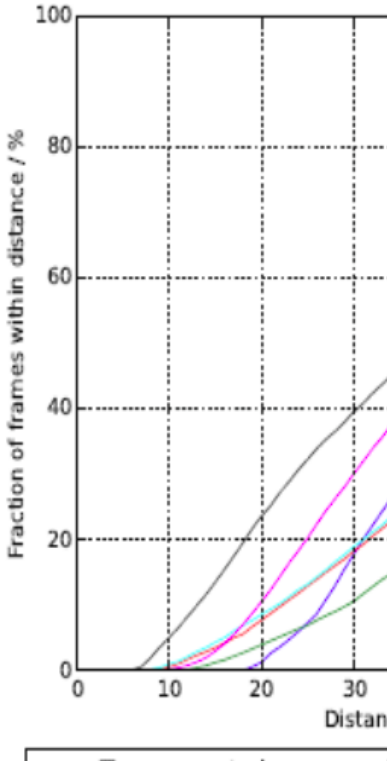


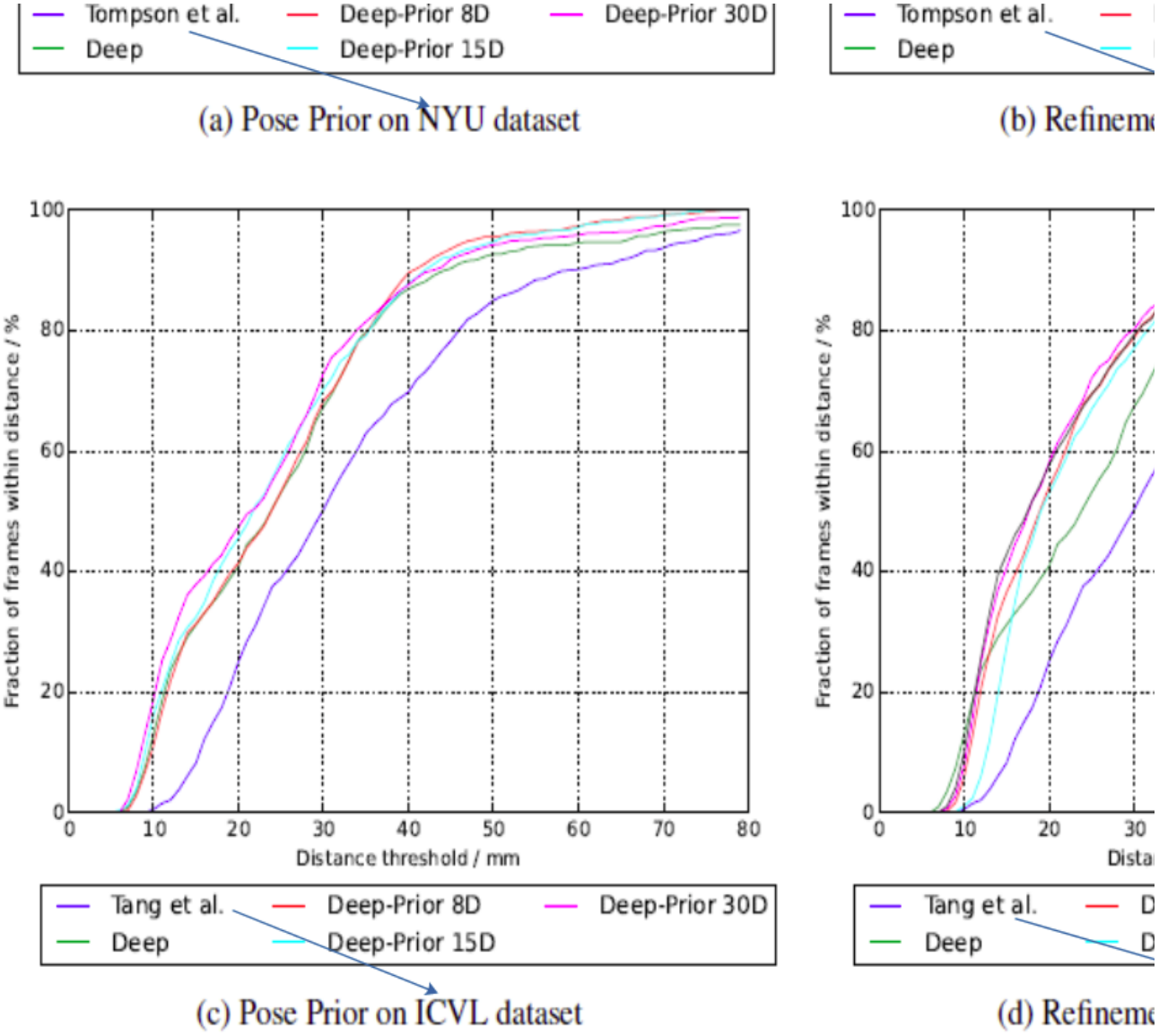


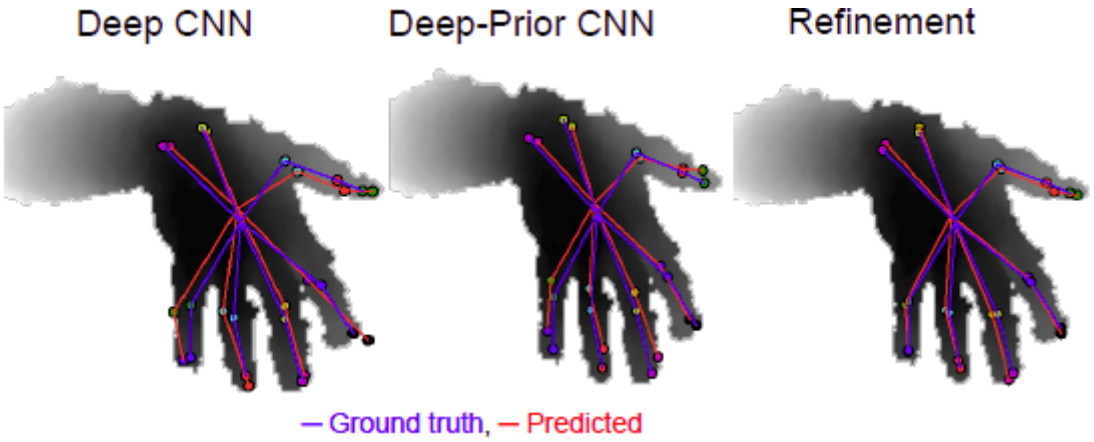
第一阶段：预测3D位置

第二阶






Tompson et al.
Deep-Prior 8D
Deep-Prior 30D
Deep
Deep-Prior 15D
(a) Pose Prior on NYU dataset
(b) Refineme
Fraction of frames within distance / %
Distance threshold / mm
Tang et al.
Deep-Prior 8D
Deep-Prior 30D
Deep
Deep-Prior 15D
(c) Pose Prior on ICVL dataset
(d) Refineme



Architecture	GPU	CPU
Shallow	0.01ms	1.85ms
Deep	0.10ms	2.08ms
Multi-Scale	0.81ms	5.36ms
Deep-Prior	0.09ms	2.29ms
Refinement	2.38ms	62.91ms
[Qian CVPR'14]	-	40ms
[Tompson ToG'14]	5.6ms	-
[Tang CVPR'13]	-	16ms

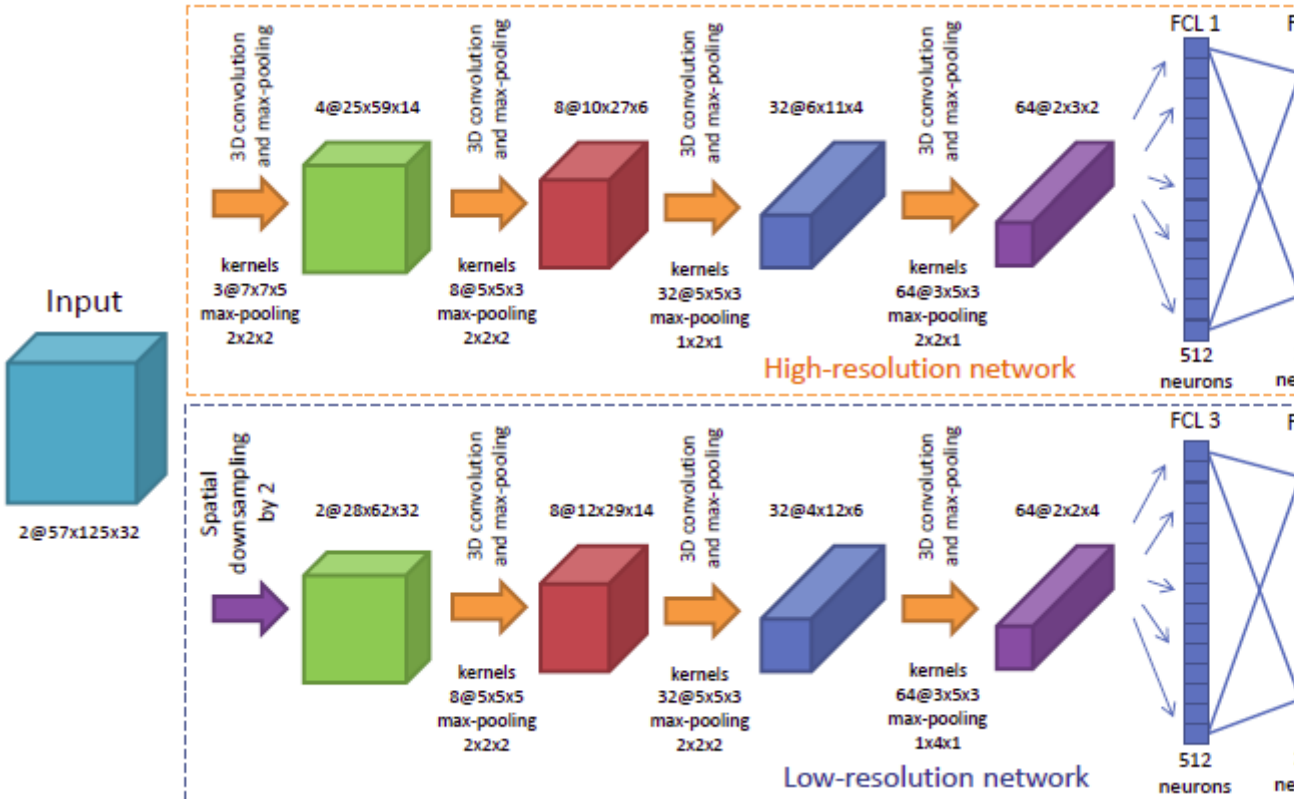
Not optimized!

【4】(2015.x)Training a Feedback Loop for Hand Pose Estimation
Markus Oberweger， Paul Wohlhart， Vincent Lepetit

- 使用Feedback Loop来纠正预测错误
- Feedback Loop也是一个深度网络，通过训练数据优化
- 避免把3D模型拟合到输入数据，不需要手工创建3D模型
- 在单GPU上可执行400fps

【5】(2015.4) Hand Gesture Recognition with 3D Convolutional Neural Networks
Pavlo Molchanov, Shalini Gupta, Kihwan Kim, and Jan Kautz

- 从深度图像中识别驾驶员手势 (NVIDIA)
- 组合多空间尺度的信息进行最后的预测
- 也利用空间-时间方法进行数据扩增，以避免训练时的过拟合
- 正确率 77.5%，基于VIVA challenge dataset



【6】(2015.5) Depth-based hand pose estimation: methods, data, and challenges.

James Steven Supan♉ci♉c III ◁ Gregory Rogez ◁ Yi Yang Jamie Shotton Deva Ramanan

- 基于单一深度帧实现了一系列的手势识别，并且发布了相关软件和评估代码
- 在单手场景中，姿势估计基本能解决
- 许多方法使用不同的标准评价自己，使比较变得困难，从而定义了一个评价标准

- 介绍了一个“简单的近邻基线”，它超越了大部已经存在的系统，表明大部分系统泛化能力差

- 强调一个未被关注的关键点：训练数据与模型本身一样重要

- Code and Dataset

- 引用【1】、【3】

Dataset	Chal.	Scn.	Annot.	Frms.	Sub.	Cam.	Dist. (mm)
ASTAR [51]	A	1	435	435	10	ToF	270-580
Dexter 1 [42]	A	1	3,157	3,157	1	Both	100-989
MSRA [33]	A	1	2,400	2,400	6	ToF	339-422
ICL [45]	A	1	1,599	1,599	1	Struct	200-380
FORTH [28]	AV	1	0	7,148	5	Struct	200-1110
NYU [47]	AV	1	8,252	8,252	2	Struct	510-1070
KTH [30]	AVC	1	0	46,000	9	Struct	NA
UCI-EGO [35]	AVC	4	364	3,640	2	ToF	200-390
Ours	AVC	10+	23,640	23,640	10	Both	200-1950

Challenges (Chal.): A-Articulation V-Viewpoint C-Clutter

Testing data sets: We group existing benchmark testsets into 3 groups based on the overall **challenges** addressed - articulation, viewpoint, and/or background clutter. We also tabulate the number of captured **scenes**, number of **annotated** versus **total frames**, number of **subjects**, **camera** type (structured light vs time-of-flight), and **distance** of the hand to camera. We introduce a new dataset (**Ours**) that contains a significantly larger range of hand depths (up to 2m), more scenes (10+), more annotated frames (24K), and more subjects (10) than prior work.

Dataset	Generation	Viewpoint	Views	Size	Subj.
ICL [45]	Real + manual annot.	3rd Pers.	1	331,000	10
NYU [47]	Real + auto annot.	3rd Pers.	3	72,757	1
UCI-EGO [35]	Synthetic	Egocentric	1	10,000	1
libhand [50]	Synthetic	Generic	1	25,000,000	1

Method	Approach	Model-drv.	Data-drv.	D
Simulate [23]	Tracker (simulation)	Yes	No	Ini
NiTE2 [32]	Tracker (pose search)	No	Yes	Ini
Particle Swarm Opt. (PSO) [28]	Tracker (PSO)	Yes	No	Ini
Hough Forest [51]	Decision forest	Yes	Yes	Dec
Random Decision Forest (RDF) [20]	Decision forest	No	Yes	
Latent Regression Forest (LRF) [45]	Decision forest	No	Yes	
DeepJoint [47]	Deep network	Yes	Yes	Dec
DeepPrior [26]	Deep network	No	Yes	Scan
DeepSegment [12]	Deep network	No	Yes	Scan
Intel PXC [18]	Morphology (convex detection)	No	No	Heuri
Cascades [35]	Hierarchical cascades	No	Yes	Scan
EPM [53]	Deformable part model	No	Yes	Scan
Volumetric Exemplars	Nearest neighbor (NN)	No	Yes	Scan

Table 4 Summary of methods: We broadly categorize the pose estimation systems `approach`: decision forests, deep models, trackers, or others. Though we focus on single-fram by providing them manual initialization. `Model-driven` methods make use of articulated ge `data-driven` models are trained beforehand on a training set. Many systems begin by `detect` or a scanning window/volume search. Finally, we made use of public source code when avail ourselves, verifying our implementation's accuracy on published benchmarks. 'Published' in results were used for evaluation, while 'public' indicates that source code was available, allo additional testsets. We report the fastest speeds (in FPS), either reported or our implement

【7】(2015.4) Fast and Robust Hand Tracking Using Detection-Guided Optimization

Srinath Sridhar, Franziska Mueller, Antti Oulasvirta, Christian Theobalt

- RDF (Randomized Decision Forest、Gaussian mixture representation)
- 50FPS without GPU support
- Website
- 没有与其它方法的比较结果，其准确度不比Deep-Prior好

【8】(2015.4) Cascaded Hand Pose Regression

Xiao Sun, Yichen Wei, Shuang Liang, Xiaoou Tang and Jian Sun

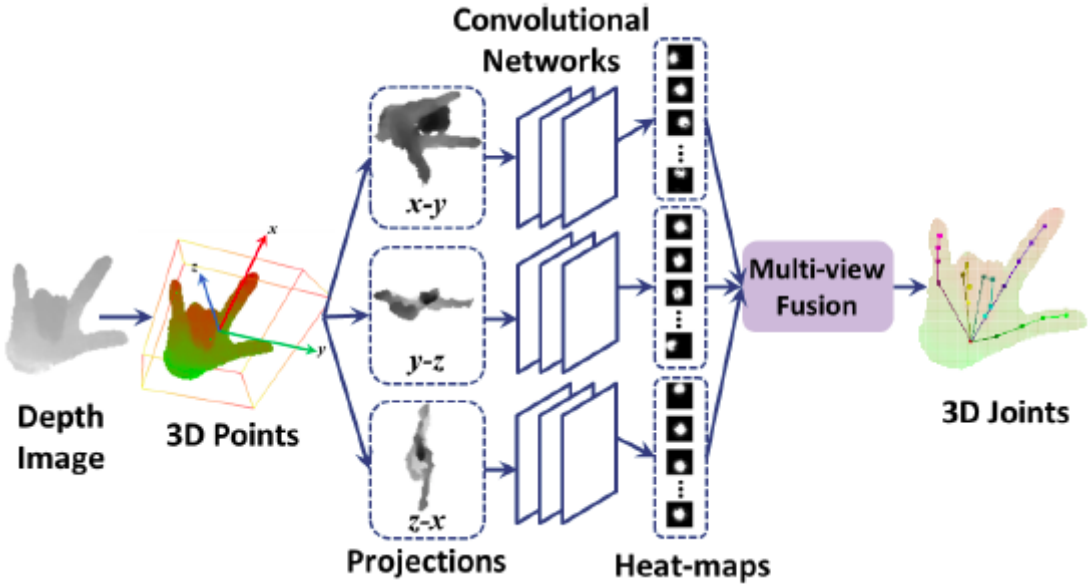
- 三维姿态索引功能（3D pose-indexed features）
- 分层回归（Hierarchical Regression）

- Dataset and Video

【8】(2016.3) Robust 3D Hand Pose Estimation in Single Depth Images: from Single-View CNN to Multi-View CNNs.

Liuhao Ge, Hui Liang, Junsong Yuan, and Daniel Thalmann

- 首先把深度图投影到3个正交平面，在每个平面上回归可以估计关节位置的热图（heat-maps）
- 把三个平面的热图融合起来，产生最后的3D位置估计，并学习先验姿势（pose priors）
- Video
- 引用【1】、【2】



【9】(2016.4) Online Detection and Classification of Dynamic Hand Gestures with Recurrent 3D CNN.

Pavlo Molchanov, Xiaodong Yang, Shalini Gupta, Kihwan Kim, Stephen Tyree, Jan Kautz

- 在真实世界中自动检测并分类动态手势的挑战：

1）人做手势时存在大量的多样性，导致检测和分类困难

2）系统必须实时在线工作，以避免用户做手势与分类结果出来之间有明显的延迟（实际需要在用户做完手势之前完成分类）

- 从多种数据中，使用递归三维卷积神经网络同时执行动态手势的检测和分类

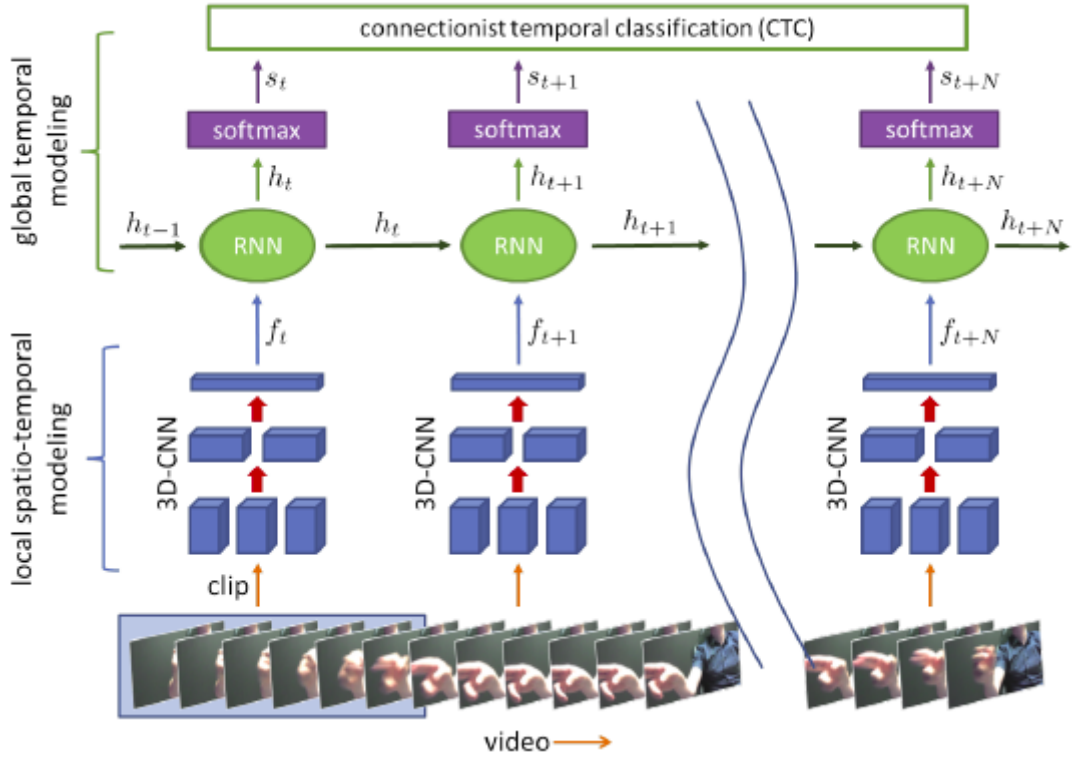
- CNN可接入多路输入数据

- 检测分类器（Detection Classifier）：区分是否有手势

- 识别分类器（Recognition Classifier）：误别出具体的手势类型

-

- Video



【10】(2016.6) DeepHand: Robust Hand Pose Estimation by Completing a Matrix Imputed with Deep Features.

Ayan Sinha, Chiho Choi, and Karthik Ramani

- 在普通计算机上可达到32FPS，无需GPU加速

- 提供了一个完整矩阵（Matrix Completion）方法，此方法在每一帧的基础上估计关节角度参数

- 深度图-->CNN-->低维特征向量 (在训练时，按此方法生成特征数据库；在识别时，使用NN算法从特征数据库中找到最近的特征向量，从而获取关节角度参数<为了加速，会记住前面帧最近邻居的特征向量>)

- 创新的矩阵算法（matrix completion algorithm）使用空间、时间最近的特征向量及其已知的姿势参数来估计输入特征向量的姿势参数

- 包括大视角的特征数据库和分层的姿势参数估计，可以解决部分遮挡的问题

- 此方法可以灵活地使用或不使用时间信息，这样大大减轻了明显的姿势初始化（当跟踪丢失或手消失时）

- 如果把深度神经网络中分类层直接换成回归层，其目标函数将陷入局部最优

- 在全连接层的倒数第二层，训练几个ConvNets输出一个判别式的低维激活特征

- 主要观点：由一系列附近的激活特征生成一个激活特征，可以更好地表示手势

- ConvNets自动学习训练范围（全局或局部）、手指类型（thumb, index, middle, ring, little）、遮挡（通过输入姿势参数的离散值），且不需要其它额外信息

- 把训练数据输入ConvNets，ConvNets输出激活特征，然后把与每一个训练图像对应的激活特征存入“激活特征数据库”中

- 主要贡献：

  1）姿势矩阵初始化（使用全局方向或手指关节的低维、差别式表示）

  2）使用一个有效的矩阵方法估计关节角度参数

  3）采用分层（全局回归、局部回归）的方法进行手势估计

- 此方法类似回归思想，但其完整矩阵方法中的“深度激活特征”与“强加的时间一致性”一起可以抑制抖动

- 此方法与协同过滤模型（collaborative filtering model）共享关系

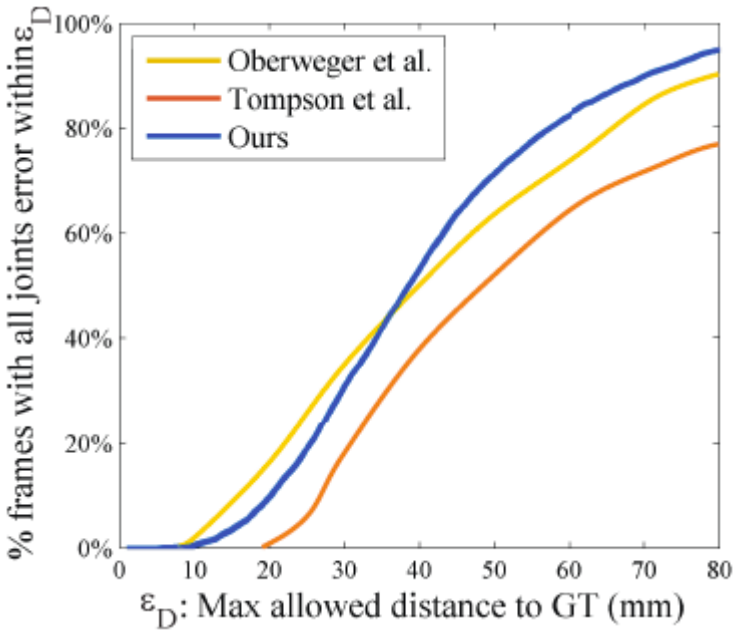
- 卷积网络（ConvNet）：不适合做回归任务，但分类任务做得很好

- 使用ConvNet计算的“激活特征”用于分类，而不是回归；把“激活特征”输入给用于实现回归的矩阵方法

- 如果每个关节角度参数一个ConvNet，其精度较好，但时间和内存消耗较大；如果使用一个ConvNet来训练所有的关节角度参数，其内存和时间消耗较小，但精度较差；所以本文采用两阶段分层的方案

- Video1 Video2

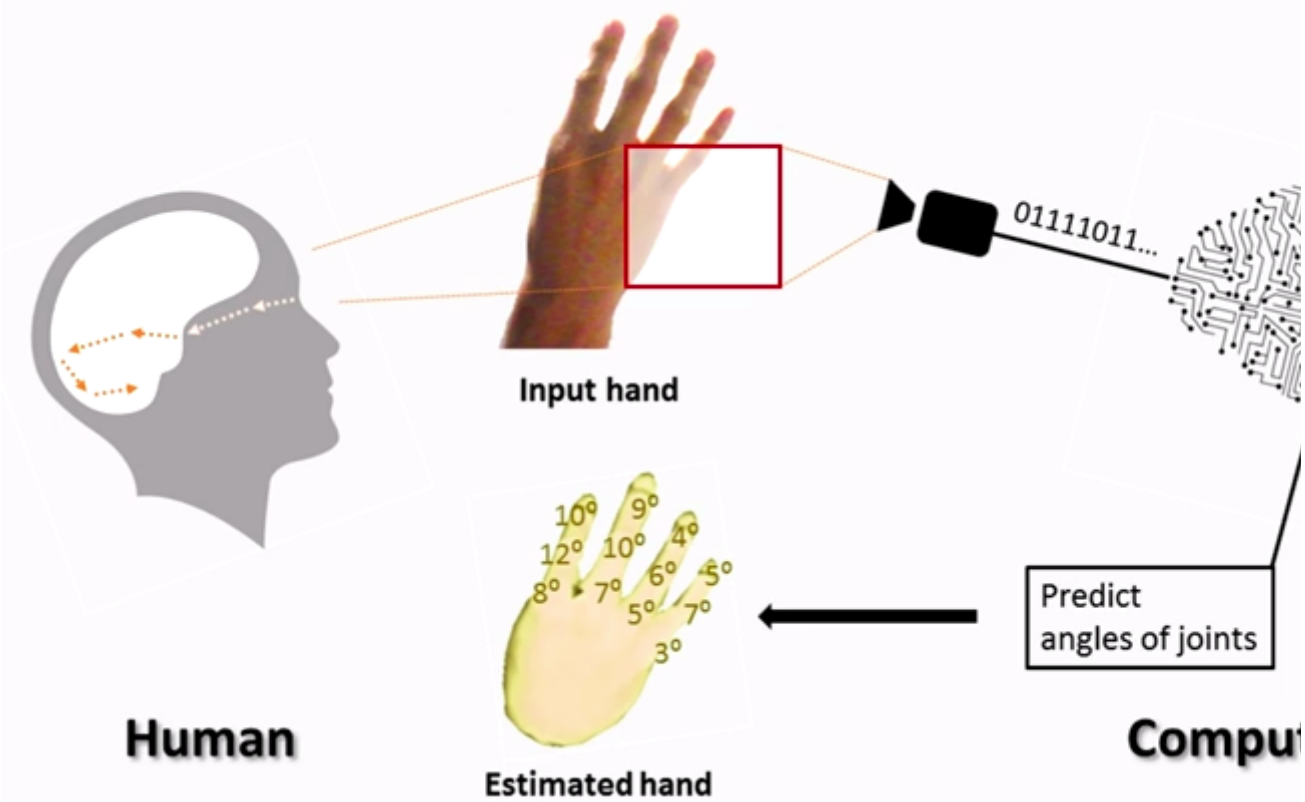
- 与其它方法的准确性比较，从图中可以可以，它的准确性并不比【3】Deep-Prior好 (GT：Ground Truth)

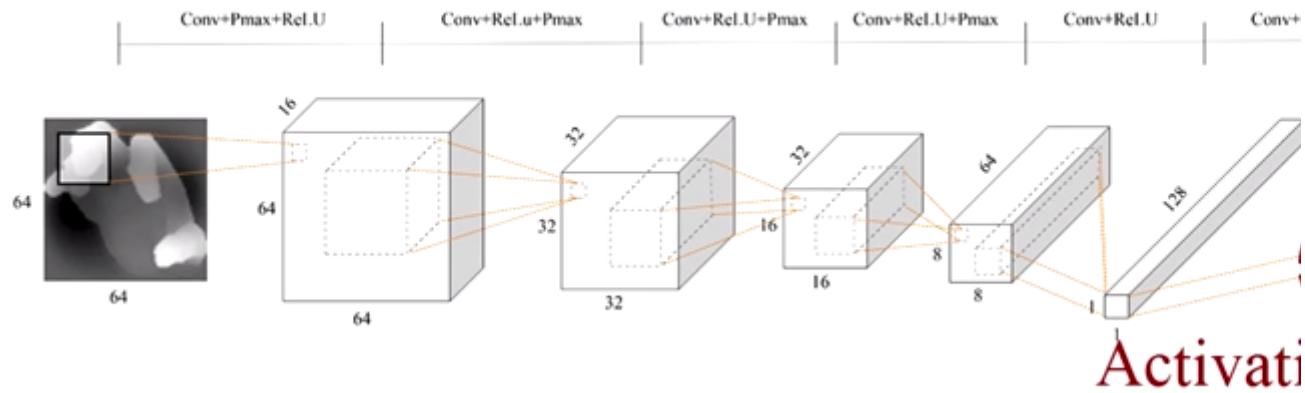


- 各种方法内存消耗比较

Model	Accuracy	Memory	Settings
RF	57.45 %	1.30 GB	22 Depth, 70 Trees
	59.04 %	1.87 GB	22 Depth, 100 Trees
ConvNet	71.01 %	2.12 MB	20 Epochs
	72.30 %	2.12 MB	25 Epochs
PCA	5.72 %	None	

Table 3: Accuracy and memory comparison of global pose initialization.


01111011...
Input hand
Predict
angles of joints
Human
Estimated hand
Compu


Conv+Pmax+ReLU
Conv+ReLu+Pmax
Conv+ReLU+Pmax
Conv+ReLU+Pmax
Conv+ReLU
Conv+
64
64
16
64
64
32
32
32
16
16
64
8
8
128
1
Activati

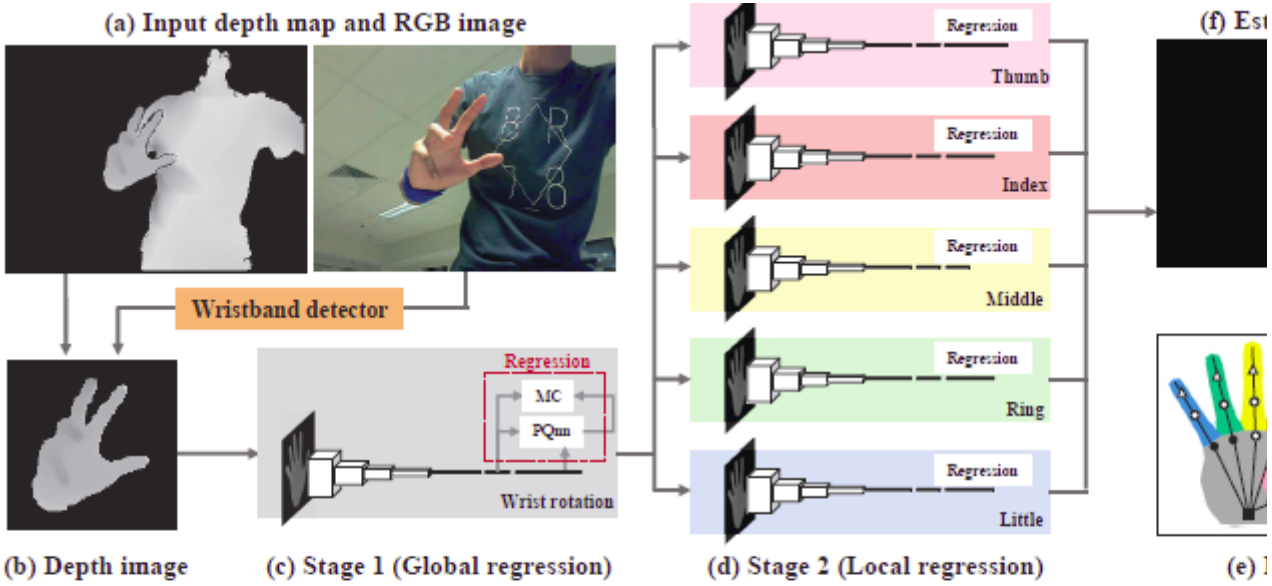


Figure 1: An overview of the proposed approach. In a real-setting, we extract region of interest using dep based wrist band detector (a)-(b). The obtained depth image is fed into a ConvNet which outputs an This activation feature synchronizes with other features in a population database using our matrix comp the global pose parameters are estimated(c). Based on this global pose initialization, we estimate the res parameters in the same recursive manner (d). The final hand pose is displayed on a multimedia screen (f).

【11】(2016.5) Efficiently Creating 3D Training Data for Fine Hand Pose Estimation.

Markus Oberweger, Gernot Riegler, Paul Wohlhart, Vincent Lepetit

- 提供了一个半自动标定深度视频帧中手关节3D位置的方法，此方法只需要用户提供参考帧中可见关节的二维投影即可。

- 利用空间、时间和形状限制获取完整序列中手的所有关节位置

- Code and Dataset

-

【12】(2014.9)Hand segmentation with structured convolutional learning

Natalia Neverova, Christian Wolf, Graham W. Taylor, Florian Nebout

- 数据对于现代算法来说有时候要高于算法本身，尤其是大面积推广深度学习以后，因此数据的多样性对模型最终精度和稳定性提供了一定的保障。但是庞大的数据标定却是一个非常痛苦的事情。尤其是网络越大，需要的数据就越多，动辄几十上百万，对于classification分分类估计还可以标一标，但是对于segmentation来说，要像素级别的标定上百万张图片，那就是不可能的

- 但是对于手势识别这样的变化非常大的任务来说，要想handle尽可能多的情况，样本会需要的更多。因此如何解决数据标定是一个难题

- 目前的解决方法是：使用合成数据，从微软的Human Pose那篇文章展现了合成数据的强大优势以来，合成数据的确是一个不错的选择，使用3D模型，天然精确标定，然后仿照现实中的情况添加一些噪声，然后就可以得到大量的标定数据

- 通过合成数据训练的模型提取context信息从而完成自动标定unlabelled样本

- 在训练的时候就用这样的方法处理了未标注的数据，那么在进行之后的训练时可以直接使用这些数据

-

【13】(2016.6) Hand Pose Estimation through Semi-Supervised and Weakly-Supervised Learning

Natalia Neverova, Christian Wolf, Florian Neboutc, Graham W. Taylord。

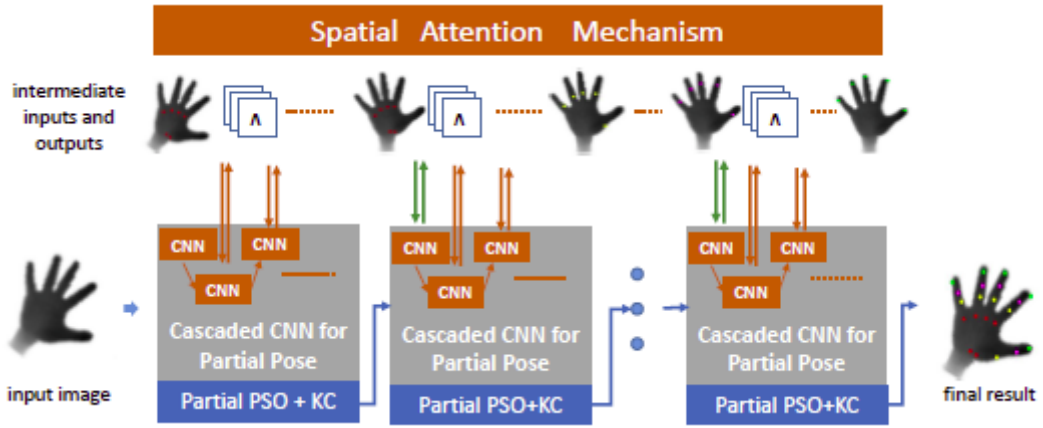
-

- 引用【12】

4. 混合方法（Discriminative methods + Generative methods）

【2.1】(2016.4) Spatial Attention Deep Net with Partial PSO for Hierarchical Hybrid Hand Pose Estimation
Qi Ye, Shanxin Yuan, Tae-Kyun Kim

- 判别方法产生的手势难以满足运动约束，而生成方法可用于纠正（或验证）这些手势结果
- 空间注意机制：它通过变换输入空间（特征空间）和输出空间的方式，把级联和分层回归集成到CNN框架中，从而大大减少了视角和关节的变化
- 在分层的级之间，分层的PSO把运动约束施加到CNN的结果上
- 3D手势估计问题可以看作是一个变量配置问题，这些变量用于表示“手势深度图像的”手模型（Hand Model）
- 级联（多阶段）+ 分层（多层）：(cascaded (with multiple stages) and hierarchical (with multiple layers))
-
-
-
- 引用【1.1】、【1】、【2】、【3】、【4】、【6】、【7】、【8】



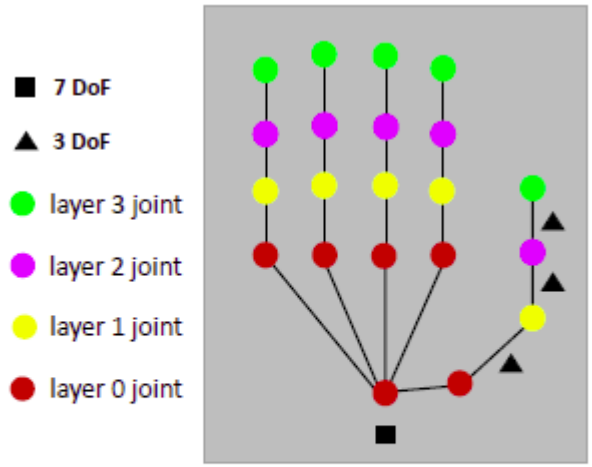


Fig. 2. Hand model. 21 joints are divided into four layers.

上图有52个自由度（DOF），layer 0有7个DOF（全局方向：一个4维单位四元数），全局位置（3DoFs），layer 1、layer2、layer3分别有15个DoF，每3个DoF表示此关节相对于上一层的旋转。

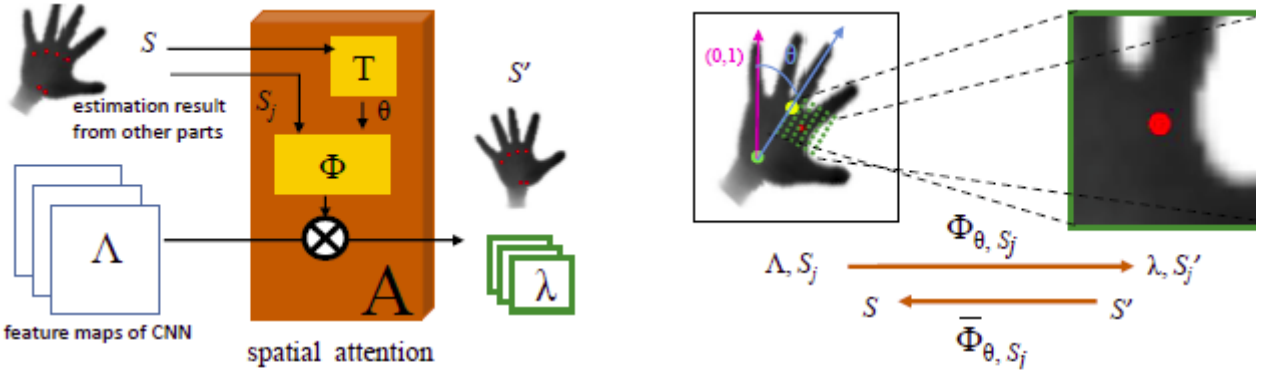


Fig. 3. Spatial attention mechanism. Left: the spatial attention module is split the calculation of rotation T and the spatial transformation Φ. Right: the mapp between input feature maps and output features maps. For clarity, we use hand ima to represent the feature maps. Both the feature maps, estimation results (or gro truth in training) are transformed to a new space by Φ_{θ,S_j}. The locations can transformed back by the inverse function $\bar{\Phi}_{\theta,S_j}$.

分类: CV

« 上一篇：CV：face recognition
» 下一篇：CV：SIFT