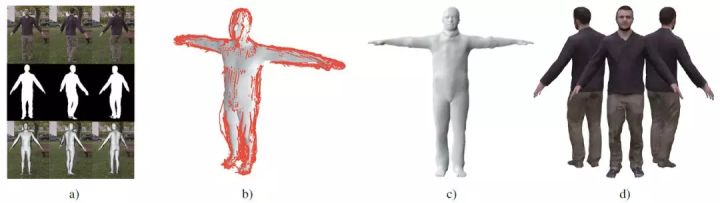
论文分析和学习

第一篇论文

概述

本论文提出的技术。首创从人的单个视频序列中提取精确的 3D 人体模型，包括头发和衣服。这些人在摄像机前面移动，从而保证我们能从各个角度看到他们。



本论文提出方法概览。输入是具备对应分割的图像序列。研究者首先使用 SMPL 模型计算姿势 (a)。然后研究者 unpose 轮廓相机光线（图中红色）(b)，并按照标准 T 型姿势优化对象的外形 (c)。最后，研究者计算纹理，生成个性化的混合形状模型 (d)。

方法

给定描绘一个移动人物的一部单目 RGB 视频，目标是生成对象的个性化 3D 模型，包括身体外形、头发、衣物、个性化的纹理图和潜在地控制表面的骨架。因此新姿势的柔性曲面变形完全是骨架驱动的。本研究提出的方法由 3 个步骤构成：（1）姿势重建 (Sec. 3.2)；（2）一致形态评估 (Sec. 3.3)；（3）帧精炼和纹理图生成 (Sec. 3.4)。本文的主要贡献在第二步，第一步建立在以前的研究上，第三步中的纹理获取和时变性细节是可选的。

该方法的关键在于将可视外壳方法泛化至单目视频（其中的人处于运动状态）。标准可视外壳方法从多个视角捕捉静态形状。穿过图像轮廓点（silhouette point）的每束相机光线对 3D 人体形状带来约束。为使可视外壳方法适用于动态单目视频，必须「撤销」人类动作，使其变成标准帧。

SMPL模型的偏移

本论文中，（穿宽松／紧身衣服的）人的几何形状被表示为标准 T 型姿势的裸体人 SMPL 参数化人体模型的偏移，该模型的特征是依赖姿势的非刚性曲面蒙皮。

SMPL模型输入是10种身形参数（β，包含人的高矮胖瘦和头身比）和72种姿态参数（θ，包含运动位姿以及24个关节相对角度）。输出是N=6890个顶点的triangulated mesh。

**T**：身形和姿态的变形。

**W**：一个混合蒙皮（blend-skinning）线性方程，其包含：

* 骨骼关节J(β)
* 与姿态有关的变形Bp(θ)
* 与身形有关的变形Bs(β)

因为SMPL是裸的人体模型，而该实验最后的目标并不是一个裸露的人的模型，而是包含衣服，头发甚至纹理，所以在T的表达式中加了一个补偿项D：



研究者首先拟合 SMPL 模型和 2D 检测，来估计每一帧的初始人体形状和 3D 姿势。基于拟合，研究者将每一帧中的轮廓点与人体模型中的 3D 点结合起来，根据每个投影光线的对应 3D 模型点的反向变形模型，变换投影光线；研究者将该操作成为 unposing。为所有帧 unpose 光线之后，研究者获得了一个可视外壳，以标准 T 姿势展示整个人体形状。之后，研究者优化人体形状参数和自由曲面顶点位移，以最小化 3D 模型点与 unposed 光线之间的距离。从而基于 SMPL 高效优化单个位移面，一次性拟合所有帧，这要求只能在内存中存储一个模型。研究者的方法首次实现从单个运动人类视频序列（这样我们可以从各个侧面看到这个人）中提取出准确的 3D 人体模型（包括头发和衣服）。

在多个 3D 数据集上的研究结果表明本论文提出的方法可以在单目视频很模糊的情况下重建 3D 人类形状，准确率高达 4.5 mm（甚至高于真值姿势的 3.1 mm）。

姿态重建

计算每个帧的3D姿势。输出是成序列的F帧的一个pose集合。

为了避免尺度模糊（scale ambiguity），每次计算时只考虑5帧，即一个身形的五个姿势。

首先，以SMPL模型的身形参数β0为固定的初始化值，为了更精确，计算时引入轮廓项。为进一步获得极小值，在高斯金字塔G的四个层次上进行了优化。



接下来使用二维关节检测和以SMPL中单模态的A姿态先验的方法来计算其姿势。用SMPL中pose参数的来训练这个prior。即进行一系列的迭代，第一帧的初始化值就是SMPL中的A姿态，而在预估新的一帧中的姿态时，初始化值是上一帧的最终预估值。如果误差过大则将pose参数重新初始化为0。除此以外，该试验方法的创新点是在计算时引入轮廓项，并在高斯金字塔的四个层次上进行了优化。最终得到的结果是一段时间内的一系列姿态。

一致形态评估

对于上一步得到的一系列姿势，从中分批给定一组估计的姿势，可以用它们联合优化单个精确的身形来匹配所有原始的姿势。因为在每一帧人的轮廓都会给身形带来新的限制，具体来说，从相机到轮廓点的光线集合定义了一个约束锥。该实验就是将顶点v的变形函数（SMPL函数）进行逆变换运用到轮廓锥的每一条光线r中。对于每一条射线r，都能找到它最近的3D模型点。对模型点对应的射线r进行逆变换。这样就能有效地消除轮廓锥，并对标准T形的身形设置约束，每一批次训练120个轮廓，经过计算得到的光线形成了一个规范的T型模型。这一步骤叫做**unposing**, 它使得身体形状得到有效优化并独立于姿态。

具体地，该实验构建了一个能量式，即1个数据能量和3个正则化能量的线性组合。

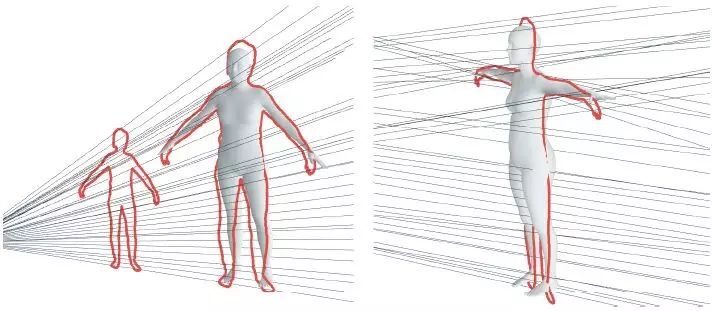
* 第一项: 轮廓项，测量边界点和轮廓线之间的距离
* 第二项：拉普拉斯项（用来保证形变的平滑性）
* 第三项：身体模型项（进行偏差处罚，即误差消减）
* 第四项：对称项（约束用来保证形体的对称性）。通过身形参数β以及补偿项D的改变可以使得整个式子的值达到最小从而得到精确的身形。



于是，一致形态被高效优化，以最大化地解释每帧实例中的轮廓。

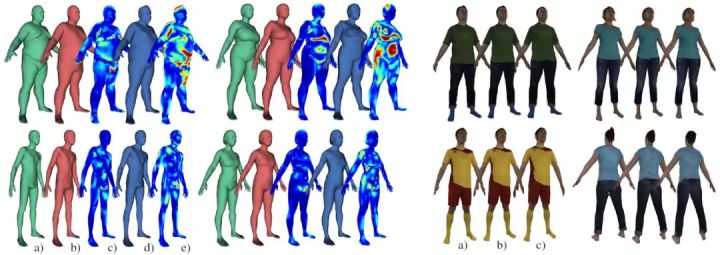
帧精炼和纹理图生成

由于时间变化导致的衣物变形，这些姿势的一致形态可能和帧轮廓有轻微的错配。因此，为了计算纹理和捕捉时间演化细节，第三步中将用滑动窗口方法对一致形态的偏离进行每一帧的优化。给出精炼的逐帧形态，我们可以计算纹理图。 本文的方法依赖于图像的前景分割。因此，研究者使用了One-shot video object segmentation（参考地址 <http://www.vision.ee.ethz.ch/~cvlsegmentation/osvos/>）中基于 CNN 的视频分割方法，并对每个序列用 3-4 个手工分割图像进行训练。为了在单目 3D 人类形态重构中克服模糊性的问题，研究者使用 SMPL 身体模型 作为起始点。



形成图像轮廓的摄像头射线（左）从无姿态形成标准的 T-姿态（右）。这种方式可以让我们在单个模型上对多个帧进行有效的优化。

实验验证



在 BUFF 和 D-FAUST 数据集的图像序列上的结果。左边展示的是 DFAUST 结果：（a) 真值 3D 扫描；（b）真实姿态的一致外观 (consensus-p）；(c）consensus-p 热图；（D）一致外观（consensus）；(e)consensus 热图（蓝色意味着 0mm，红色意味着 ≥ 2cm）。右边展示了在 BUFF 数据上的结果（a) 真值扫描；（b) 真实姿态的一致外观；（c）一致外观。



定性测试结果：因为重构的模版与 SMPL 人体模型有同样的拓扑结构，我们可以使用 SMPL 来改变重构形体的姿态和外形。虽然 SMPL 不建模带衣着人体的变形，但变形模版看起来还是可信的，从质量上来看足够承担多种应用。

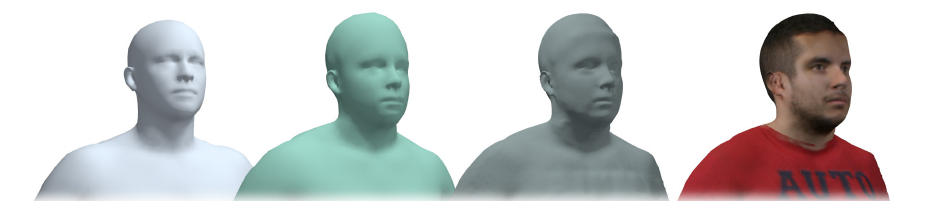


输入图像（竖线左）与构建形象（竖线右）的对比。正如图像右侧所示，用本文方法构建的形体准确地覆盖了输入图像。在右下角还呈现了构建模型的侧面视角。

参考资料：<https://zhuanlan.zhihu.com/p/35750579>

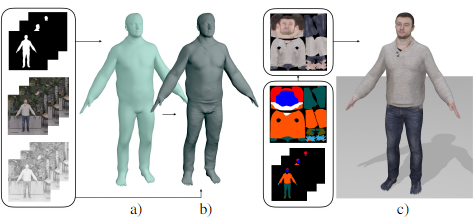
第二篇论文

概述



本论文提出的方法。从一个人的单眼视频中创建一个详细的化身。 基于SMPL模型，首先计算中级头像，然后添加主题特定细节并最终生成无缝纹理

基于上一篇论文提出了一种基于单目视频的保留了高细节的虚拟人物创建方法。对参数化的人体模型进行了改进和优化，使其与视频中的各个角度显示的受试者都极为相似。这种模型的特点是自然的面孔，发型，衣服与服装褶皱，以及高分辨率的纹理。该实验在几个重要方面改进了最先进的技术，即他们的优化方案能够实现集成人脸地标检测和来自多个帧的阴影形状。除此之外，该实验还引入了一种新的纹理拼接的二元优化方案，它包含了每个语义分割部分的外观模型。



本文的方法概览。研究者首先根据分割（a）估计中等水平的身体形状，然后使用阴影形状（b）添加细节，最后使用语义先验和新颖的图切割优化策略（c）来计算纹理。

方法

对纹理网格模型进行了从粗到细的重构，重构过程由 3 个步骤构成：（1）中等体型重建 (Sec. 3.2)；（2）建模精细的表面细节 (Sec. 3.3)；（3）纹理生成 (Sec. 3.4)。

SMPL 模型的改进

首先，为了最后能得到一个具有更多细节和更明确个人特征的模型，该实验对 SMPL 模型上的每一处边缘都进行了两次细分，每个新顶点都定义为:



其中定义形成边的顶点对，是顶点对的法线之间的平均法线。这样就得到了新的顶点和更加精细和有着光滑表面的模型。这是基于第一篇论文改进后的 SMPL 模型所进行的进一步改进。

中等精度体型重建的改进

在上一篇论文的第二步共识身形估计中的能量式中新引入了一项 Eface（脸部能量项），来进行脸部细节的优化：



Eface 项抵消（penalize）二维人脸地标检测与三维人脸地标的二维投影之间的距离。这是三维和二维之间的连接方式。该实验还通过OpenPose（参考地址<http://www.doc88.com/p-0476379515159.html>）来检测二维地标。为了将检测结果引入到该方法中，该实验在网格上建立了地标和点之间的静态映射。每个地标l都通过邻近顶点的重心插值映射到表面。在优化过程中，该实验测量了模型上的地标 l 与对应的摄像机射线 r之间的点线距离，从而描述了 unposed（去姿态的）空间中的二维地标检测：



精细表面细节的建模

仅仅从轮廓估计得来的身形模型是粗糙的。为了得到更加精确的结果，该实验提出了基于阴影来恢复形状的方法。

因为织物的运动和错位，从全部帧的信息来估计 shape-from-shading 会得到一个平滑的没有任何细节的身形。因此，首先首先单独地捕获多个关键帧的细节，然后，随着新的三角形在连续的关键帧中慢慢变得可见，将这些细节逐渐融入到模型中。经过验证，发现每批选择的关键帧数小于第一步的中等精度体型重建，改为 60 帧（之前是 120 帧）。

**Shape-from-shading:** 对于每一帧图像，我们首先使用基于 CNN 的本质分解方法[3]将图像分解为反射率项 Ir 和阴影 Is。函数 Hc 计算具有球面调和分量（spherical harmonic components）c 的顶点的阴影。对于给定的帧窗口，通过使模拟阴影与观测图像阴影之间的差异最小化，可以估计得到球面调和分量 c：



有了场景照明度和每个像素的阴影，该实验开始估计每帧每点的辅助法线： 

**表面重建:** 为了融合窗口内所有估计的法线信息,该实验使用姿态函数的逆函数 SMPL M^-1 来将法线转变回到标准的 T 字形的模型上。接着该实验对含有合并法线信息的表面进行优化。除此之外，该实验还添加了第二步的轮廓项和表面项 Esilh和 Eface 进来。总的来说，估计这一关键帧的精细细节就是需要使得一个能量式最小化，得到经过优化计算后的补偿项 D 以及沿着 unposed 的模型的点的法线方向上的位移集合 s。具体能量式里面是 60 帧的每一帧的轮廓项和表面项以及Shape-from-shading 项



和正则化项



的线性组合：



其中，正则化项是四个能量项的线性组合。第一项Ematch 项抵消了两个相邻关键帧之间的差异；第二项 Elap 项是具有各向异性权重的拉普拉斯平滑项；第三项Estru 项（相关论文参考地址http://www.doc88.com/p-954212822723.html）的目的是通过修剪边缘长度的变化来保持网格的结构；第四项Econs项删除了与共识形状的偏差较大的偏差。

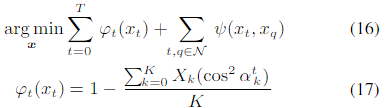
该实验使用了 dog-leg 信任区域方法，使用 chumpy 自微分框架进行优化，并且该实验交替最小化并找到轮廓点与直线的对应关系。其正则化是逐步减少的。

纹理生成

高质量的纹理图像是一个真实的虚拟人物必不可少的组成部分，因为它可以描述不能由表面几何建模的材料属性。为了获得清晰且无缝的纹理，该实验解决了每个纹素层面上的纹理拼接。换句话说，该实验的目标是实现纹理图像中的每个像素都是用从 K 个关键帧中提取的一个像素值为其着色。这会使得问题的规模更大，从未不能实现全局优化。为此，该实验提出了一种基于图切割的纹理合并方法，该方法将纹理合并问题转化为一系列可以有效求解的二值标记子问题。此外，我们还可以看到一个问题，网格和关键帧没有完全对齐。为了减少颜色溢出和不对齐造成的伪影，该实验在拼接最终纹理之前先计算了语义先验。

**部分纹理生成：**对于每个关键帧，该实验首先将所有可见的表面点投影到该帧上，并将投影位置的颜色写入相应的纹理坐标。为了剔除纹理图像中的照明因素，该将输入图像与之前 3.3 节中使用的阴影图像分开来，从而对其进行解阴影处理。接着，利用 OpenGL 栅格化管道可以方便地实现局部纹理计算。除了部分彩色纹理图像外，该实验还为合并步骤计算了两个纹理图像，即视角映射和语义映射。对于视角映射，该实验计算了关键帧中曲面点所对应的视角。

**语义先验：**将人类的语义分割重新投影到纹理空间中，就生成了语义先验。具体来说，该实验首先使用基于 CNN 的人工解析方法[5]计算输入帧中每个像素的语义标签。每一帧都被划分为 10 个语义类，如头发，脸，左腿和上衣。然后通过最小化标签 x，将所有帧的语义信息融合到全局语义图中。

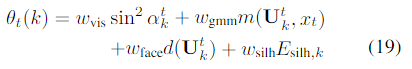


在构造图形的过程中，该实验不仅将每一个纹素与纹理空间中的相邻纹素连接起来，还将其与表面上的所有相邻纹素连接起来。这意味着纹素与纹素间是跨纹理接缝连接的。为了让纹理补全有较强的先验，该实验利用部分纹理和相应的标签计算 HSV 空间中每个标签颜色的高斯混合模型(GMM)。

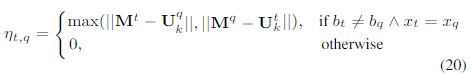
**纹理合并：**接下来，该实验通过合并这些部分纹理来计算完整的纹理。在保持图形结构不变的情况下，目标函数为:



u将每个纹素分配给部分纹理k。 第一次迭代旨在为每个纹素找到最佳图像：

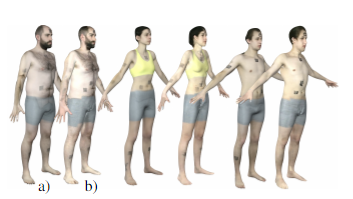


由于η中的组合数非常高，因此在计算上不可解方程。η近似有：



通常在 2K 到 3K 次迭代之间达到收敛。最后，该实验交叉混合不同的标签，以减少可见的接缝。该实验使用了使用标准图形切割程序库，使用 Python 代码的 1000 \* 1000px 上的每次迭代运行时间为 2 秒。但是该实验未进行运行时间优化的尝试。

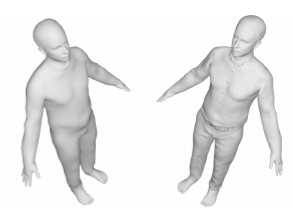
实验验证



我们的结果（b）与RGB-D方法[9]（a）相比较。注意这里尚未使用纹理优先级。



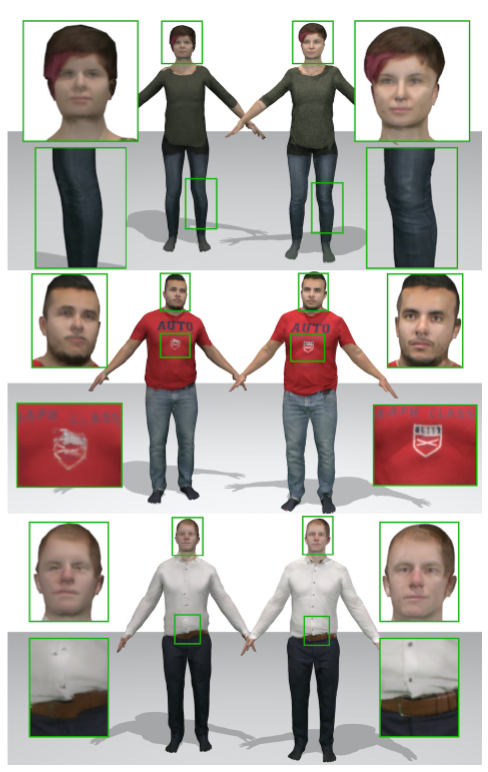
与[3]（左）的方法相比，我们的结果中的面（右）在网格中具有细节，并且与照片中的主体非常相似。



比较我们的方法之前（左）和之后（右）应用基于着色的形状细节增强的结果。



纹理拼接的语义先验成功地消除了我们最终纹理中的颜色溢出（左）（右）。



与[3]（左）的方法相比，我们的结果（右）看起来更自然，并且具有细节。