指导教师：杨涛　　提交时间：2015.03.29

The task of Digital Image Processing

数字图像处理

School of Computer Science

No：1

姓名：王海燕

学号：2012302570

班号：10011208

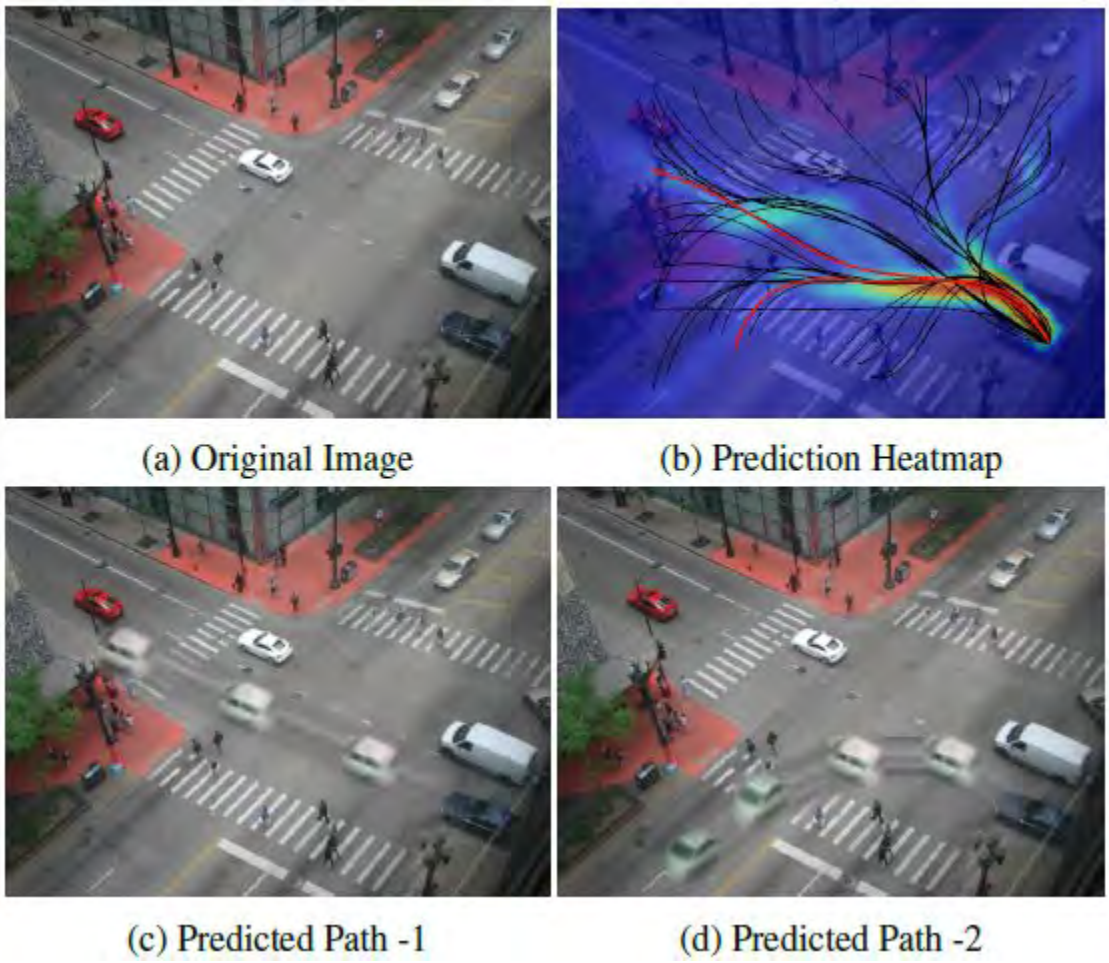
Patch to the Future: Unsupervised Visual Prediction

Jacob Walker, Abhinav Gupta, and Martial Hebert
Robotics Institute, Carnegie Mellon University
{jcwalker, abhinavg, hebert}@cs.cmu.edu

摘要

在这篇论文我们提出了一种视觉预测算法，它将中层视觉元素的有效性与时序建模结合起来，概念简单但强大惊人。我们的框架，可以通过从大量的视频之中通过完全无监督学习得到。然而，更重要的是，正因为我们的预测框架是建模在这些中层视觉元素上的，所以我们的方法不仅可以预测场景中可能出现的运动，还能够预测视觉表象——即表象将如何随着时间的推移而改变。算法将在屏幕上生成一些可能发生的事件的虚拟的预测图像，我们用“幻像”来称呼这些图像场景。你们可以看到

我们的方法能够作出准确的预测，并能可视化简单的未来事件。同时，我们还表明，这种非监督学习的方法，在预测事件上，可以与监督性学习的方法相媲美。



(a) Original Image (b) Prediction Heatmap

(c) Predicted Path -1 (d) Predicted Path -2

图表 1. 请注意图(a)中的场景。我们的数据驱动方法正在使用大量的视频集来预测场景中智能体的未来。热图(b)显示了汽车未来可能访问的位置(伴随这一些可能的路径.) (c)显示了汽车径直向前开的幻象而且

(d)汽车左转的幻象.

1、导言

大家看图 1，我们知道，在现代，一个可靠的机器视觉的算法充其量只能够识别图片中的一些特定的物体和区域，并把它们对于相应的类别之中，如：道路，车辆，树和草等。然而，人类不仅能够通过看到的场景图像推断当时的情况，甚至还能预测场景的后续事件。举个例子，在图 1 中我们可以预测：右下角那两车，当它到达十字路口时，它只会向前直走或者左转。而人类的这种惊人的观想未来的能力主要是来自我们已有的对视觉世界的丰富知识。

视觉预测为何如此重要？原因有两点：(a)对于智能体和智能系统来说，预测是决策的关键。例如，为了执行辅助活动，机器人必须要能够预测场景中其他智能体的意图。即使步行穿过拥挤的走廊这样简单的人物，也要求智能体能够预测人类的运动轨迹。(b)更重要的一点，视觉预测需要深入理解视觉世界，并能与场景中的不同元素进行复杂的交互。因此，从某种方面来说，我们可以认为“理解图像”即指通过图像对即将发生的事物进行预测。图像 1 的任务即是预测，如图（a）我们的算法通过大量的这样的视频的数据驱动来预测即将发生的场景。热图（b）中显示了右下角黑色车辆未来可能访问的地点（以及几条可能的轨迹），

（c）图显示了小车直走行驶的幻象图，(4)图显示了汽车左转行驶的幻象图。视觉预测可以作为“场景理解”的试金石。

在这项工作中，我们将比一般的视觉预测——识别活跃物体以及预测其接下来的活动——更进一步。这就带来一个很重要的问题，我们应该预测什么？我们的视觉预测的输出结果如何表现出来？最近大家的算法都致力于用点[18]来表示智能体，用光线[33]来表示智能体的运动和转换。相比之下，我们人类不仅能够预测智能体的运动，还能够预测在其运动时其形态（在视觉中）的变换。这让我们能够在脑海中创立一副当前环境的预测图像。类似的，我们认为视觉预测的空间应该更加丰富，甚至能够预测视觉中智能体表象的变化。例如，我们可以猜到小车转向时样子，能够预测一本书缓缓开启的样子。然而，更加丰富的输出就意味着我们需要更加丰富的用以推理的表示元素以及大量的数据来习得这种先觉能力。在当前比较火的中层元素[28]基础上，我们提出了一个新的视觉预测的框架，在框架中，我们把这些中层元素作为构建块来进行预测。我们的框架不仅能够对场景中元素的运动以及转换进行建模，还能将预测元素外观的变化。我们的新框架具有以下优势：（a）我们从不假定场景中有哪些是智能体，而是直接通过大量的数据驱动来对可能的智能体进行识别。 （b）这种以块为单位的表示元素使得我们可以用一种完全无监督方法得到视觉预测的模型；同时，丰富的表示元素使得我们的可以用简单的无参数算法习得最先进的视觉预测模型。（c）最后，

因为我们方法用的是中层元素而非整个场景，所以这些模型具有普遍性，且可以用于不同的实例之中。

1.1 背景

预测是智能的一个重要的组成部分[13]，甚至对于老鼠和鸽子这样的智能体也是如此[34]。神经科学领域的研究者为人类大脑中的感官预测提供了大量的学术支持[2]。而当预测在生物视觉领域被广泛研究的同时[1]，计算机视觉领域则大都专注于对研究场景或视频中语义[7,27,32]和几何知识[10,14]的理解和推测。最近，一些研究者开始致力于对人与场景[11],[15]和物体[9],[31]的交互进行静态功能建模，以在之后用于对场景与人之间的交互进行预测。然在，在这项工作中，我们只关注暂时的、短时间的预测，我们的目标是预测接下来将会发生什么。

这种短时间的预测大致有两种方法。第一种是无参数法，这种方法依赖与大量的数据[33]而且不是对智能体和环境进行假定猜想。例如，[33]索引与输入相似的场景，然后据此对预期的移动建模。然在，这种基于场景的匹配，需要海量训练数据，因为你需要对世界中的物体根据所有可能的时间、空间设定进行明确的建模。因而，最近的研究大都专注于用基于 warping 的方法[20]对相似却不完全一致的场景进行预测。

第二种截然相反的方法就是：参数化建模方法。这种方法中，人类

猜测场景中有哪些活动元素，他们也许是汽车，也许是人；一旦假定其为活动元素，就构造一个模型来预测智能体的行为[35]。基于追踪的方法大都专注于预测智能体在短时间范围内的行为，因此多使用线性模型[17]。而对于较长时间的预测，常用的有：Markov Decision Process (MDP) [18, 35], Markov Logic Networks [29], ATCRF [19], CRF [8], 和 AND-OR graphs [12, 25] 等。这些预测模型的推理过程涉及到了从规划到制定决策的各种方法[18,35]。然而这些手段却有一些缺点：(a)他们做了太多的假定性工作 （b）这些方法人仍旧依赖于语义分类这种至今都还很能解决的问题。最后，（c）这些方法都需要明确的去选择活动物体，如汽车或人类。在大多数情况下，人们使用人工监测或者对象探测（一种很鲁棒的技术）的方法来训练模型。不过，我们使用的是数据驱动的方法，并且我们的预测框架基于很容易检测和追踪的中层元素。这使得我们可以用完全无监督的方式来训练我们的模型。但是更重要的是，丰富的视觉表示代表元素，使得我们可以同时预测外观的变化。

在这项中做中，结合这两种动态场景建模方法，我们提出了另一种概念简单但功能惊人强大的方法。进来，“发现中层可识别块[4,5,6,16,28]”的研究非常成功，我们即在此基础上提出了基于这些中层元素的预测框架。也正是因为我们的框架基于这些中层块，所以我们的方法比那些基于场景匹配的方法基于更小范围的，更好的一般性。然而，我们并非只是将中层块与训练数据匹配，我们还对这些元素块用基于上

下文的马尔科夫模型进行建模。我们不仅对这些元素块的运动以及外观变化建模，还可以通过学习上下文模型（或者类似于[18]中的 reward function）来捕捉其与场景之间的关系。例如，一辆车（car patch）一般不会在人行道上行驶，因为如果这样做将会付出很高的代价（high cost），但是一个人（people patch）却很有可能在人行道上移动，我们直接利用图像的特征建立这种：场景——元素块 的上下文模型，而非中间语义层（上文提到的那个很难解的问题，语义分析）。你可以看到，我们的学习方法在跟踪误差上的鲁棒，以至于可以通过完全非监督的方式习得。

2 我们如何实现

对于一副输入场景，我们需要预测接下来会发生什么——图像中哪部分将会保持不变，哪些部分可能发生移动，它将怎么移动。我们的中心思想，是用那些可能发生移动或外观变化的智能体的中层元素集（通过滑动窗口探测得到）来代表整个场景。我们假定场景是静止的(包括其他智能体)，对每个智能体进行独立的预测。我们通过过渡矩阵，对分布在空间中的各种可能的行动进行建模（过渡模型），这些模型表示了中层元素将如何移动和转换成另外的中层元素，以及各种移动和转换的概率。比如，一个代表正向行驶的汽车的元素块，在汽车转向之后，将变成一个向右行驶的汽车元素块。根据这些中层元素块以及所有他们可能发生的行动，第一步，我们可以决定谁最有可能是智能体，其次，

根据给出的场景判断哪种行动最有可能发生。但是，哪种行动可能发生，不仅取决于我们的追踪目标，还取决于目标周围的场景/上下文。例如，在图 1 中，汽车的视觉预测就不只与其本身的关，还与图像中的其他车辆，行人，人行道等有关。因此，第二步，我们需要对智能体与其周围环境的交互进行建模。我们用回报函数$\psi_i(x,y)$对这种交互关系进行建模。$\psi_i(x,y)$的值代表了类型为 i 的元素块将移动到地点（x，y）的可能性。例如，一辆车的元素块对像道路的区域有很好的回报值，而对像草地的区域有较低的回报值——没有明确的语义建模。给定一个追踪目标，我们的算法即会用过渡矩阵给出可能的路线，并同时计算出回报值（section 2.4）。最后，假如这是一个未知目标——如这里给出的案例，我们建议抽样出几种目标，并且选择预期回报率最高的一种。

在训练中，我们需要学会：（a)中层元素表示； (b)当前空间和每个元素可能发生的过渡模型；（c）每个元素的回报函数$\psi_i(x,y)$。我们将从大量的时空视觉数据中，通过非监督学习的方法得到这些结果。首先，我们创建一个包含中层块的静态空间域，将这个域与其余的视觉世界区分开来，比如在一个表示在道路上移动的车辆的视频集（一个域），或者到户外散步的行人的视频集（一个域）。我们首先用[28]的方法，从这些域中提取出中层元素块。这些元素都是从大量普通的视频数据中训练得到的有意义且有区分性的 HOG 聚簇。每个元素，都可能表现为一个

可以移动的智能体。我们的方法通过挖掘数据来判定哪些特征非常重要并抽象出智能体，而不是基于整个域去设想智能体是人类还是车辆。举个例子，在 VIRAT dataset[23]的案例中，一个元素包含了两个行人，因为这两个人非常可能一起移动，所以可以被建模成一个单独的智能体。一旦我们从给定的域中抽象出了中层元素“字典”，我们就利用时空信息，和图片平面的空间信息（Section 2.1）来得到块到块的过渡关系。我们用过渡矩阵的统计信息来断定哪些元素块代表了场景中的智能体。最后，我们根据静态空间算出回报函数，再与过渡矩阵结合起来进行预测。

2.1 学习过渡

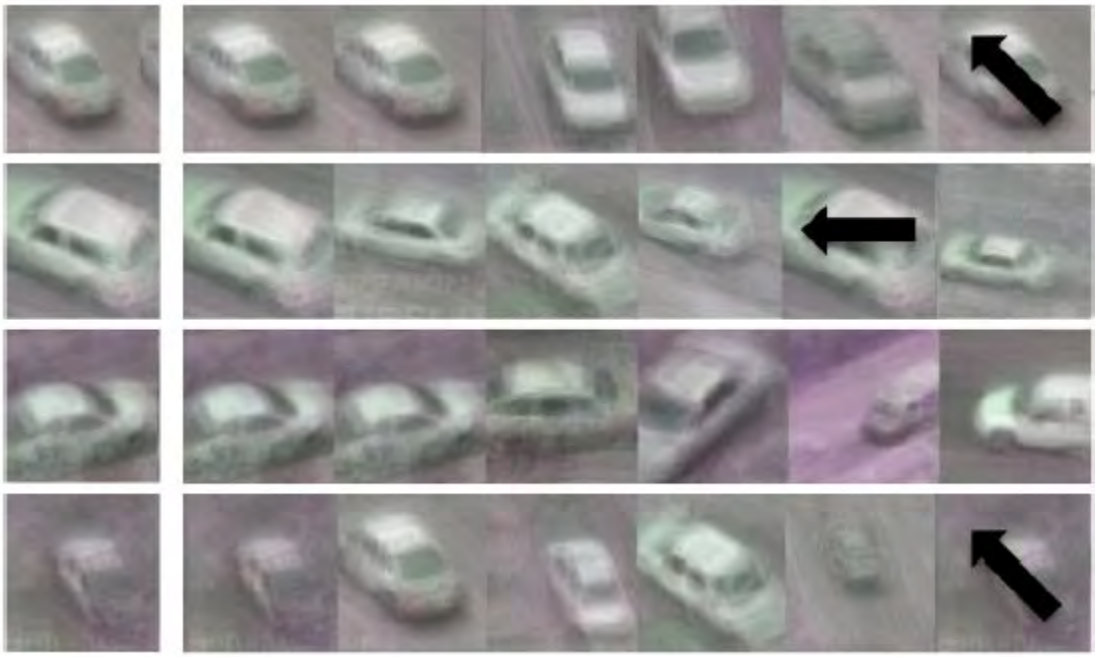
给定中层元素“字典”，第一步是建立这些元素的时序模型。这个时序模型可以由一个过渡矩阵表示，表名元素块 i 可能向 8 个方向中的某一方向移动（上、左、下、右、左上、右上、左下、右下），或者转变成为另一个中层元素块。我们如何习得这些过渡呢？给定一个训练集，

我们以至少一秒一对的速度抽取出多双帧图，检测到其中的中层块。为了习得过渡关系，我们需要得到两幅帧中检测结果的对应关系。我们使用 KLT Tracker[22]来对两个相邻框（boxes）中相同的特征进行计数以得到块的对应关系。我们把这些对应关系对映射到时间或者空间的变换

之中。如果两个块属于不同的聚簇，则不管其空间运动如何，都把这次匹配作为一个聚簇之间的转换。在图片平面上，如果要把元素块计数为空间移动，两个相匹配的元素块必须是同一类的，并且两者之间不能重叠（见图 2 中的例子）。为了补偿相机的移动，这些块的运动都是在通过 SIFT[21]匹配生成的全景图上计算得到的。对每一个变换，我们也都规范了观测得到的元素块的总数。这使我们可以得到对于中层元素块每种变换的可能性。图 3 显示了四种元素块最有可能的三种变换。

图表 2. 中层块匹配举例.用单映预估匹配两幅帧图，并且每幅帧图检测框边界内的 KTF 特征

指引着元素块的运动。



图表 3.从训练数据学习到的最可能的过渡。左边是原始元素，右边是可能的过渡。注意每个元素块既有可能改变成另外一种元素，也有可能只是在空间中移动(标有箭头的方块).元素被显示为最佳检测的平均图像。

2.2 学习上下文环境

一个过渡矩阵只能在不考虑当前的上下文信息（场景影响，其他智能体的影响）的情况下捕捉可能发生的运动。例如，一辆面向右边的车，最有可能向右转。然而，智能体的行动不仅仅依赖于可能的变换（上文得到的过渡模型），还依赖于其周边环境。例如，假如这辆车前面是一面墙，它就不可能向那个方向移动。因此，除了通过统计学习的方法捕捉元素块可能的过渡，我们还需要得到其与周边环境可能的交互的信息。我们用回报函数对这种交互信息进行建模，回报函数 $i(x; y)$ 代表

了图像上一个 i 类型的元素移动到地点（x,y)的可能性。因为每个元素都应该代表不同的基本概念，所以我们对场景中的每个元素都通过学习建立单独的交互回报函数。

我们用分片的方法来建模无参数的回报函数。为了得到元素类型 i 的回报函数训练数据，我们在训练视频中检测这种元素，并从时间角度观察场景中哪些片段易于与其发生重叠。举个例子，汽车元素就很可能和道路片段重合， 因此，这些道路片段对于汽车元素是积极的正回报率区域。用这种方法，我们为元素“字典”中的每个元素建立一个训练集。一旦我们得到了每个元素块类型 i 的训练集，我们就可以用这些在检测阶段计算出其回报函数。被检测图片中的每一个片段，都可以通过图片特征匹配找到自己几何距离上最近的邻居，我们选择最近的 N 位邻居标记为高回报率区域，然后根据视觉上的相似性在图片上传递回报函数值——看上去与高回报率区域相似的当前求解图像也可以得到很好的回报率。图 4 展示了一个回报率传递的例子。

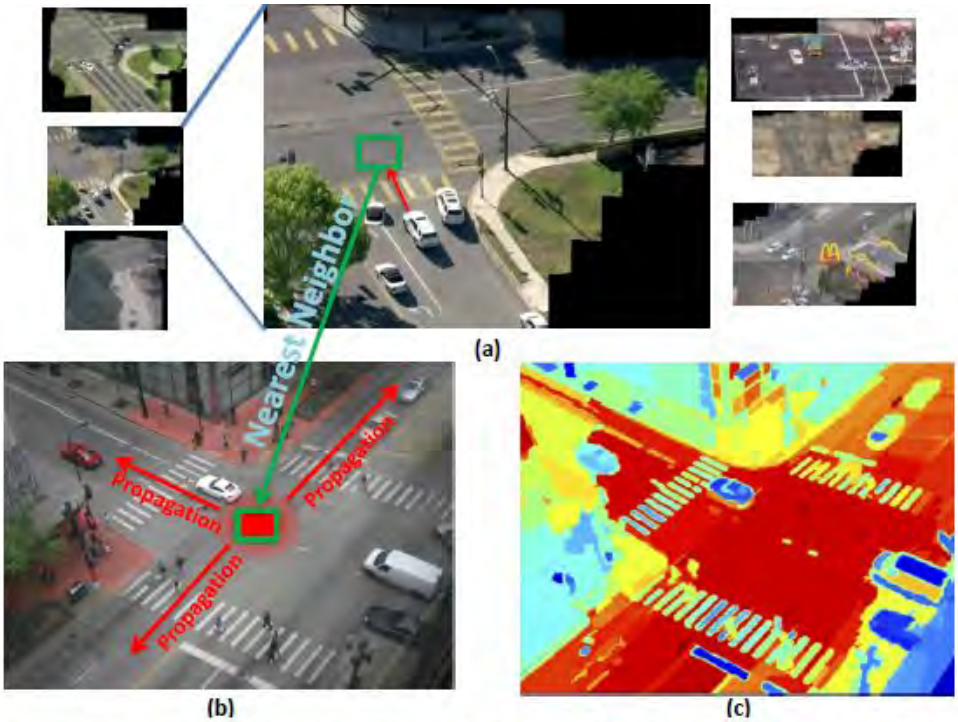


Figure 4. A reward function (c) is propagated by taking texture information from the destinations of observed moving patches in training data (a). During test time (b), the area of the image with the closest texture to the training textures is set as the highest reward in the scene. Other areas of the scene (via graphcut segments) are scored according to the similarity to the chosen window. Warm colors indicate high reward; cooler colors indicate low reward.

2.3 推断运动实体

一旦我们习得了每个中层元素的过渡函数以及回报函数 i(x; y)，我们就可以判断接下来会发生什么。预测推理的第一步是估计场景中可能的活跃元素。Kitani 以及其他人[18]手动的检测活跃智能体。在此，基于学习到的过渡矩阵，我们提出一种自动推理可能的活跃智能体的方

法。我们的基本思想是根据空间活跃的可能性大小对各种聚簇类型进行排序。我们假定活跃智能体元素满足以下条件：（a）易于移动；（b）易于转化为可移动的块；（c）处在一个允许元素块移动到其周围的高回报区域的场景中。为了检测到场景中最可能满足这些条件的元素，我们用滑动窗口检测法检测这些元素的实例。然后，根据上下文信息对这些实例进行排序。每个元素块 i，在地点（x，y）的 context—score 用如下公式表示：

$$\sum_{d} p_i^d e^{\psi_i(x+d_x, y+d_y)} \tag{1}$$

在这个式子$d=(d_x,d_y)$表示运动方向，p_i^d表示向方向 d 转变的可能性，$\psi_i(x+d_x,y+d_y)$计算出把元素块从(x,y)处移动到$(x+d_x,y+d_y)$处的回报。在论文中，我们将 d 离散为 8 个方向。

我们并非要预测所有发现元素块，而只预测那些可能直接或通过变换来改变位置的活跃元素（可以理解为过渡的两种模式，位置移动，或外观变化）。并且，我们通过过渡矩阵来计算这种改变的可能性。因此，很可能运动的块、能转变成易于运动的元素的块将会被选中。

2.4 规划动作以及选择目标

一旦我们选出了最可能的活跃智能体，给定一个场景中的空间目标，我们就可以用过渡矩阵和回报函数来搜索最佳动作/转变。我们首先将回报函数$\psi_i(x,y)$重新参数化，令$s=(x,y,i)$（元素块 i，处于地点（x，

y))，则回报为$\phi(s)$。每个可能的决策 a，都可用期望回报进行量化：即新状态下决策 p_a 的概率和回报。注意有移动和转换两种类型的决策。在第一个案例的状态下，是（x，y）的地点转变了；而在第二个案例下，地点未变，但聚簇类别改变了。

我们的目的是找到最优行为/决策集 $\sigma = (a_1, ..a_n)$，使得这些行为/决策有着最大的期望回报（最小代价），且这些行为可以达到目标状态 g。

求最大回报值的公式：

$$\max_{\sigma} \sum_{a_t \in \sigma} p_{a_t} \phi(s_{t+1}) \quad \text{s.t.} \quad \sigma \odot s_0 = g \qquad (2)$$

其中 s0 表示初始状态。⊙是一个操作符，它表示对一个状态(如 s0)执行各种行为（如 σ 集）来估计目标状态。然后，我们使将回报值转换为代价，用 Dijsktra 算法规划化出一个从最初状态到目标状态 g 的最佳决策序列 σ 。具体来说，我们创建一个曲线图（graph），每中状态都用图标中的一个节点表示。例如，对于一个 100x100 的图像且字典大小为 750 个元素，其曲线图（graph）中将有 100x100x750 个节点。两个节点之间的边代表了从状态 si 过渡到状态 sj 的代价。这个代价取决于过渡的概率和回报值。给定一个图表，初始状态由图表中的源节点代表，而目标状态则被认为处于图像的边缘。之后，我们运用 Dijsktra 最短路径算法得到最优路径。我们从众多不同的目标之中基于平均期望回报选择最佳路径——这是对于有不同决策总数的情况的规范化做法。

2.5 实现细节

KLT Tracker： 我们对提取出的 SURF 特征[3]使用 Kanada-Lucas 追踪算法，来追踪检测到的块如何在各个场景之中移动。对处于两个不同帧中的块，给定一个初始块，我们对给定块的边界框内的 SURF 特征进行追踪。

回报函数：回报函数的距离测度用一个基于 RGB 和 a bag of words 的 69 维的特征向量来计算。

其他细节：对于过渡矩阵建模学习时所选的帧，其中 VIRAT 数据集帧间间隔为 4 秒，汽车追踪数据集因为移动速度快，帧间间隔只有 1 秒。

3、实验结果

因为视觉预测领域的研究工作很少，所以没有的现成的数据集，基线和评价方法。因而我们为路径预测做了大量的定性评估和定量评估工作。

基线：当前并没有非监督视觉预测算法；因此我们和最进的邻居相比较，然后是 sift-flow warping[20,33]与 Kitani 等人基于 max-entropy 的 Inverse Optiaml Control（IOC）的算法[18]。

对于 NN（nerual network）基线，我们用的是类似于 Yuen 等人[33]的 Gistmatch[24]方法.我们把最邻近的标记路线当作预测轨迹，然后用 Sift Flow[20]将其 warp 到当前场景中。参照 Kitani 等人的方法，我们首先用 IOC 习得回报函数，然后对于一个给定的初始智能体，我们用 Markov Decision Process（MDP）预测最可能的路径。

数据集：我们用两个数据集进行实验：一个是汽车追踪数据集（从 YouTube 上收集得到），另一个是 VIRAT 数据集。

评估标准：我们用改进的[18]中的 Hausdorff distance 算法（MHD）来衡量两条轨迹之间的距离。

通过在小瞬时窗口上（在我们的实验中是三步时长）找寻最一致的局部点，MHD 允许局部时间 warping。每种算法都将生成一个可能的预测路径集合，因而我们将会计算 top-N 条生成路线与真实路面路径(ground-truth)之间的距离。

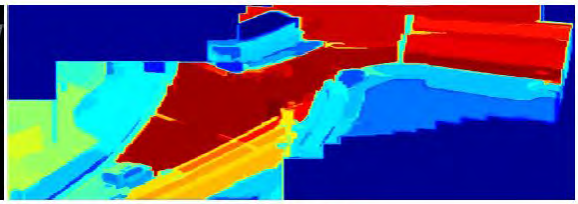
汽车追踪数据集

为了进行我们的主要实验，我们用从 Youtube 上下载的高架铁道汽车行驶视频创建了一个新的数据集。我们总共用到了 183 个时长 5-30 秒的视频（含 48 个不同场景）。其中，我们用 139 个含 37 种不同场景的视频

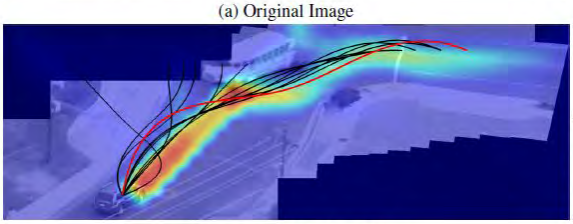
来训练，用 44 个含 11 种场景的视频作为测试集。为了抽象出不同的中层元素，我们用了 1871 张从训练集的随机抽取的帧图，以及 309 张从 Flicker 找到的户外场景图来作为元素发现数据集，还用了 MIT Indoor 67[26]数据集来作为反面数据集。我们在 44 个测试视频中手动标注汽车轨迹来当作用于算法评价的真实地面轨迹(ground-truth)。

定性：图表 5 显示了定性结果。注意看回报函数是如何捕捉到道路是汽车元素块的高回报率区域的。而图像顶部的公交车与底部的汽车则是低回报值区域。类似地，人行道也被认为是低回报值的区域。给定回报函数，我们的推理算法会生成可能的路径，并且会在图表中显示边缘化的这些路径以及一些样本路径。最后，注意看（d）和（h）中的视觉预测。注意观察图像顶部汽车为了躲避公交车是怎么转弯的，以及这辆汽车是如何从图片底部的两辆汽车之间穿过的。图表 6 展示了我们的算法生成的一些定性的视觉预测实例。

(a) Original Image

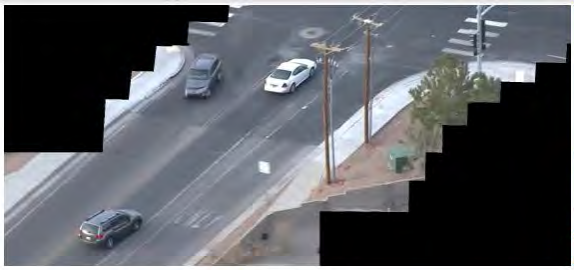


(b) Reward Function

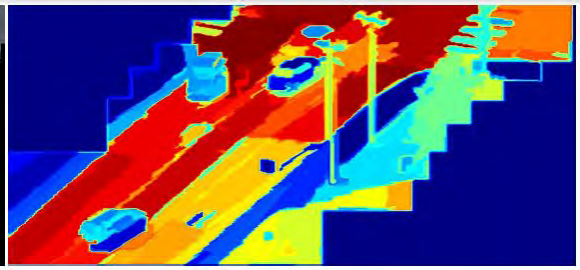


(c) Distribution of Predicted Paths

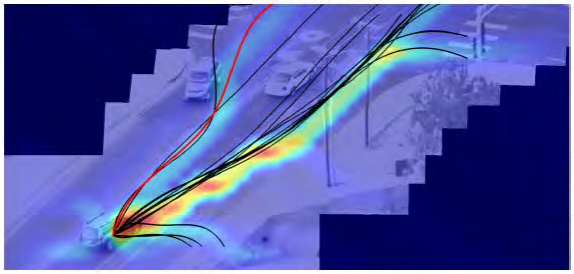
(d) Predicted Path



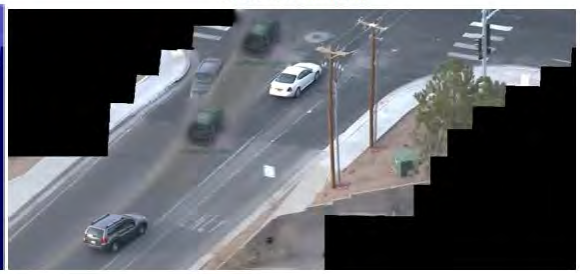
(e) Original Image



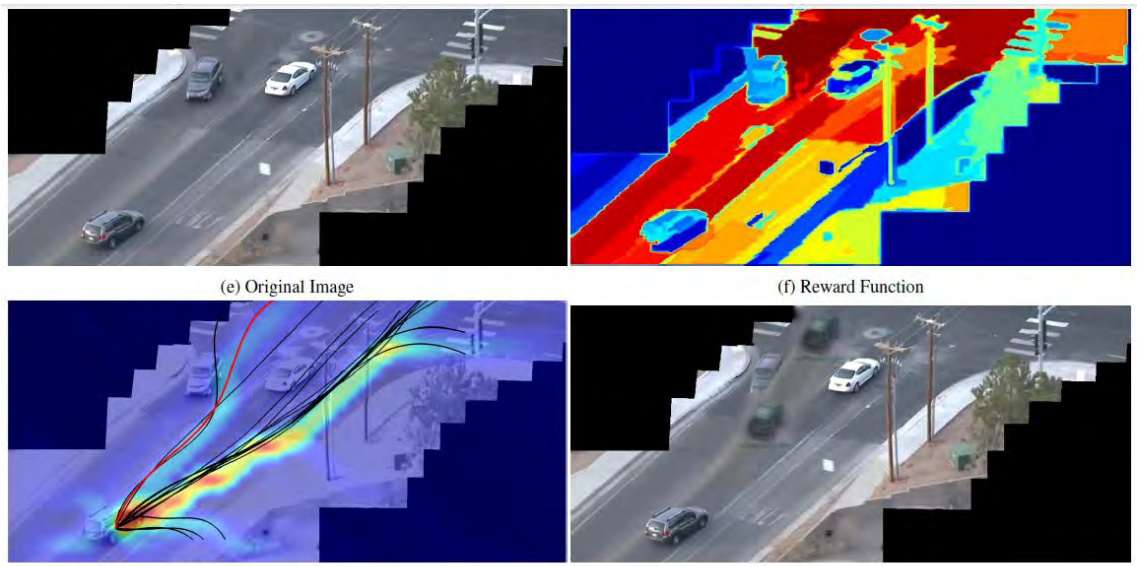
(f) Reward Function



(g) Distribution of Predicted Paths



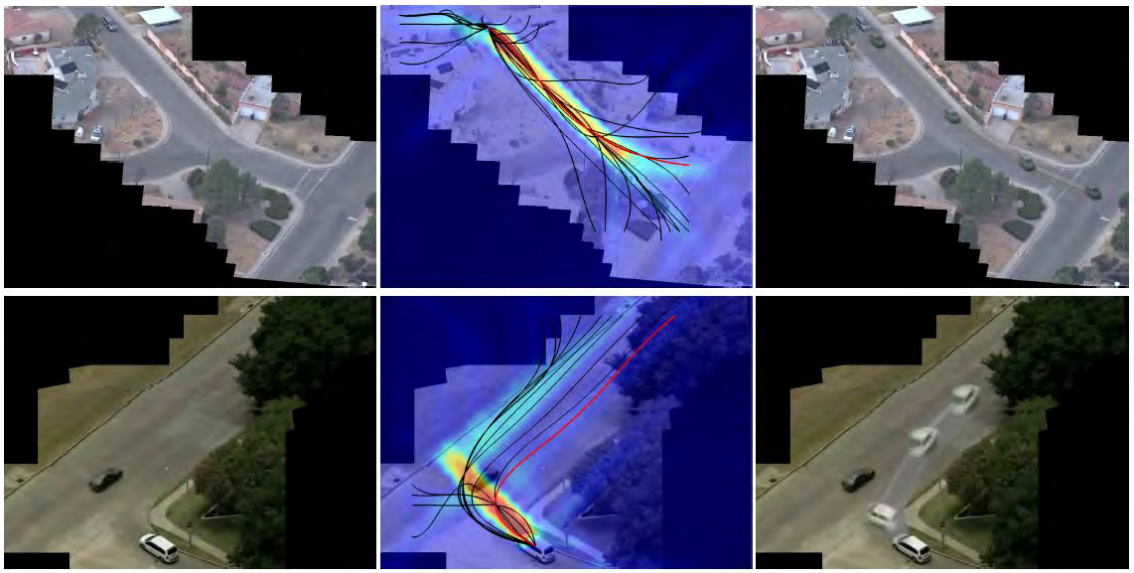
(h) Predicted Path

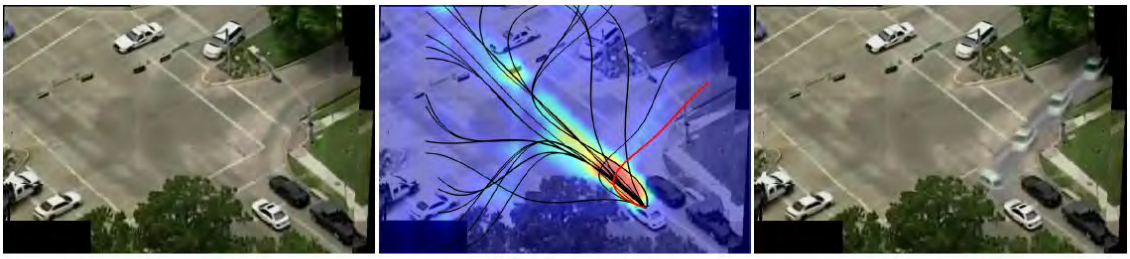


(e) Original Image (f) Reward Function

(g) Distribution of Predicted Paths (h) Predicted Path

Figure 5. Some qualitative predictions of our approach. The upper right images (b,f) represents a reward function for a given car patch over the image. The lower left (c,g) shows a heatmap of possible locations where the agent can be and some of predicted trajectories, and the lower right (d,h) demonstrates the visualization of one such trajectory.





(a) Original Image (b) Prediction Heatmap (c) Predicted Path

Figure 6. Qualitative predictions for our approach. The far left shows the original image, the center shows a heatmap of possible paths, and the right shows a visualization of one of those paths.

定量：在第一次实验里面，在没有给定智能体的情况下，我们将自己的方法和 NN-baseline 进行了对比。通过这种方法，我们对我们的算法找到场景中正确的活跃实体，以及更重要的——有效地预测其空间动作的能力进行了测量。我们用自己的算法识别了 3 中最可能的活跃智能体并且为每个智能体预测了两条路径。我们同样允许 NN 算法基于 top-6 的匹配生成六条路径。所有用到的全景图都被调整为最小 50 像素的规范尺寸。表格 1（top）展示了这种方法的表现。对于中值误差，我们在 NN 基线的基础上提升了 35%的性能。在 73%的案例中，真实地面汽车都是活跃度处于前三的实体。

Experiment	NN + Sift-Flow	[18]	Ours
No Agent Given			
Mean Distance	22.34	-	**14.38**
Median Distance	16.68	-	**10.91**
Agent Given			
Mean	27.55	37.94	**21.55**
Median	23.77	30.23	**14.98**

Table 1. Mean and Median of the error of closest path over 44 videos in the car chase dataset for no agent given, and Mean and Median error of the top-ranked path for a given agent.

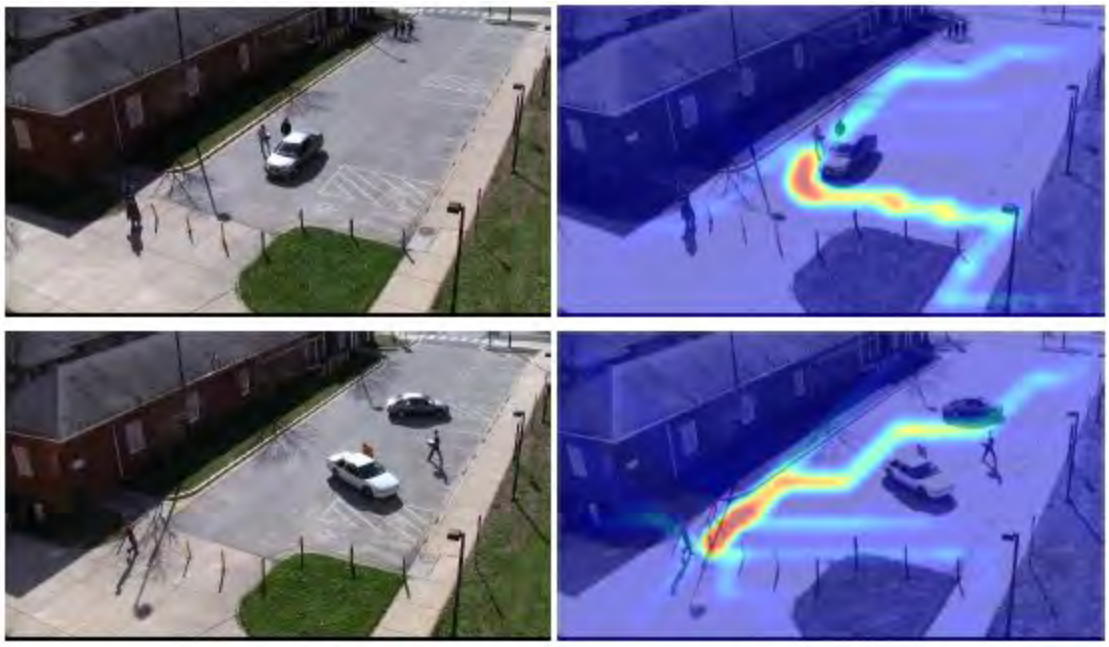
在第二个实验中，我们探寻了另在外一个问题：给定一个手动选择的智能体，我们预测所有可能的动作的分布的能力有多好？在这次实验中，我们加入了[18]来作为第二基线。路径是根据期望回报值排序的。一旦给定了智能体，我们即对案例中的最可能的 N 条路径，而不是像[18]中的全部的路径分布进行比较。表格 1（bottom）展示了我们的方法再一次优于 NN-baseline，它甚至表现得比[18]还优秀。[18]在这个案例中表现得不好是因为潜在的语义特征的影响。

3.3 VIRAT 数据集

对于我们的第二数据集，我们选择了与[18]使用的单一场景 A 一致的 VIRAT 子集。既然 VIRAT 数据集是由单一场景组成的，我们就用 TUD-Brussels 户外行人数据集中的帧图来抽取中层元素。我们依照[18]中实验设计来训练模型。我们用较[18]3 倍的交叉验证与 15 步长的窗口来执行 MHD。我们还使用了全像素网格（359x634）。然后发现，我们的方法能够比[18]更好的推测目标和路径。为了证明我们的回报函数是有意义的，我们同样与 MDP 进行了比较，只是这一次用的是回报函数而不是 IOC。表格 2 显示了这个方法的表现情况。对于这个实验，由于已经给定了智能体，我们就将这几种方法生成的最佳预测轨迹进行了比较。

VIRAT	Ours	MDP (Our Reward)	[18]
Mean	**108.81**	128.48	147.32
Median	**77.79**	99.05	150.24

Table 2. Mean and median of the error of top predicted path.



(a) Original Image (b) Prediction Heatmap

Figure 7. Qualitative predictions for our approach on the VIRAT dataset. The left shows the original image; the right shows a heatmap of possible paths.

4.结语

在这篇论文中，我们展示了一个用于静态场景视觉预测的一个简单却有效的框架。我们的框架建基于易于区分且有代表性的中层元素，并将这些视觉代表与决策理论框架结合起来。这种表示方法使我们可以用大量的视频来完全无监督地训练我们的框架。然而，更加重要的是，我们

还能够预测视觉外观将来根据时间的变化，并且创建未来场景的幻象。从经验上看，我们的方法甚至表现得比需要使用多种数据集监督性方法更为优越。有一点需要特别注意的是，这篇论文只是无监督视觉预测方向的第一步，我们仅仅在假定余下场景信息是静止的的情况下对事件进行了预测。在未来的工作中，将可能包括对多个元素的同步行为进行建模，来预测他们之间可能的协同动作。

Acknowledgements: This work was supported in part by NSF grant IIS1227495.

References

[1] M. Bar. The proactive brain: memory for predictions. Philosophical Transactions of the Royal Society, 364(1521):1235–1243, 2009.

[2] M. Bar, editor. Predictions in the Brain: Using Our Past to Generate a Future. Oxford University Press, 2011.

[3] H. Bay, T. Tuytelaars, and L. Van Gool. Surf: Speeded up robust features. In ECCV, 2006.

[4] C. Doersch, A. Gupta, and A. A. Efros. Mid-level visual element discovery as discriminative mode seeking. In NIPS,2013.

[5] C. Doersch, S. Singh, A. Gupta, J. Sivic, and A. A. Efros. What makes Paris look like Paris? ACM Transactions on Graphics (TOG), 2012.

[6] I. Endres, K. J. Shih, J. Jiaa, and D. Hoiem. Learning collections of part models for object recognition. In CVPR, 2013.

[7] P. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In CVPR, 2008.

[8] D. Fouhey and C. L. Zitnick. Predicting object dynamics in scenes. In CVPR, 2014.

[9] H. Grabner, J. Gall, and L. Van Gool. What makes a chair a chair? In CVPR, 2011.

[10] A. Gupta, A. A. Efros, and M. Hebert. Blocks world revisited: image understanding using qualitative geometry and mechanics. In ECCV, 2010.

[11] A. Gupta, S. Satkin, A. A. Efros, and M. Hebert. From 3D scene geometry to human workspace. In CVPR, 2011.

[12] A. Gupta, P. Srinivasan, J. Shi, and L. S. Davis. Understanding videos, constructing plots learning a visually grounded storyline model from annotated videos. In CVPR, 2009.

[13] J. Hawkins and S. Blakeslee. On Intelligence. Times Books,

[14] V. Hedau, D. Hoiem, and D. Forsyth. Recovering the spatial layout of cluttered rooms. In ICCV, 2009.

[15] Y. Jiang, M. Lim, and A. Saxena. Learning object arrangements in 3D

scenes using human context. In ICML, 2012.

[16] M. Juneja, A. Vedaldi, C. Jawahar, and A. Zisserman. Blocks that shout: Distinctive parts for scene classification. In CVPR, 2013.

[17] V. Karavasilis, C. Nikou, and A. Likas. Visual tracking by adaptive kalman filtering and mean shift. In Artificial Intelligence: Theories, Models and Applications. 2010.

[18] K. Kitani, B. Ziebart, D. Bagnell, and M. Hebert. Activity forecasting. In ECCV, 2012.

[19] H. S. Koppula and A. Saxena. Anticipating human activities using object affordances for reactive robotic response. In RSS, 2013.

[20] C. Liu, J. Yuen, and A. Torralba. Sift flow: Dense correspondence across scenes and its applications. PAMI, 2011.

[21] D. G. Lowe. Distinctive image features from scale-invariant keypoints. IJCV, 60(2):91–110, 2004.

[22] B. D. Lucas, T. Kanade, et al. An iterative image registration technique with an application to stereo vision. In IJCAI, 1981.

[23] S. Oh, A. Hoogs, A. Perera, N. Cuntoor, C.-C. Chen, J. T. Lee, S. Mukherjee, J. Aggarwal, H. Lee, L. Davis, et al. A large-scale benchmark dataset for event recognition in surveillance video. In CVPR, 2011.

[24] A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. IJCV, 2001.

[25] M. Pei, Y. Jia, and S.-C. Zhu. Parsing video events with goal inference and intent prediction. In ICCV, 2011.

[26] A. Quattoni and A. Torralba. Recognizing indoor scenes. In CVPR, 2009.

[27] J. Shotton, J.Winn, C. Rother, and A. Criminisi. Textonboost for image understanding: Multi-class object recognition and segmentation by jointly modeling texture, layout, and context.

IJCV, 81(1):2–23, 2009.

[28] S. Singh, A. Gupta, and A. A. Efros. Unsupervised discovery of mid-level discriminative patches. In ECCV, 2012.

[29] S. D. Tran and L. S. Davis. Event modeling and recognition using markov logic networks. In ECCV, 2008.

[30] C.Wojek, S.Walk, and B. Schiele. Multi-cue onboard pedestrian detection. In CVPR, 2009.

[31] B. Yao and L. Fei-Fei. Modeling mutual context of object and human pose in human-object interaction activities. In CVPR, 2010.

[32] B. Yao and L. Fei-Fei. Action recognition with exemplar based 2.5D graph matching. In ECCV, 2012.

[33] J. Yuen and A. Torralba. A data-driven approach for event prediction. In ECCV, 2010.

[34] T. R. Zentall. Animals may not be stuck in time. Learning and

Motivation, 36(2):208–225, 2005.