

вать не обязательно.) Теперь сделайте то же самое для выбранного вами фрагмента речи, состоящего из 5–10 предложений.

- 22.14.** Мы забыли упомянуть, что текст, приведенный в упр. 22.1, должен быть озаглавлен “*Стирка белья*”. Еще раз прочитайте этот текст и ответьте на вопросы, приведенные в упр. 22.7. Удалось ли вам на этот раз лучше справиться с заданием? Бренсфорд и Джонсон [173] использовали этот текст в эксперименте, проводимом под лучшим контролем, и обнаружили, что для его понимания очень важен заголовок. Какие выводы вы можете сделать по проблеме совершенствования речи?

23 ВЕРОЯТНОСТНАЯ ОБРАБОТКА ЛИНГВИСТИЧЕСКОЙ ИНФОРМАЦИИ

В данной главе показано, как можно использовать простые языковые модели, прошедшие статистическое обучение, для обработки коллекций, состоящих из миллионов слов, а не просто отдельных предложений.

В главе 22 было показано, что агент может взаимодействовать с другим агентом (человеком или программой), используя фрагменты текста на естественном языке. Для полного извлечения смысла фрагментов речи необходимо проводить всесторонний синтаксический и семантический анализ фрагментов речи, а такая возможность возникает благодаря тому, что эти фрагменты речи невелики и относятся только к ограниченной проблемной области.

В данной главе рассматривается подход к обеспечению понимания языка, основанный на использовании **совокупностей текстов**. Совокупностью текстов (*corpus*, во множественном числе — *corpora*) называется большая коллекция текстов, подобная тем миллиардам страниц, из которых состоит World Wide Web. Эти тексты написаны людьми и для людей, а задача программного обеспечения состоит в упрощении поиска нужной информации. В этом подходе предусматривается использование статистики и обучения для получения возможности воспользоваться содержимым совокупности, и в нем обычно применяются вероятностные языковые модели, обучение которых может проводиться с использованием существующих данных и которые проще по сравнению с дополненными грамматиками DCG, описанными в главе 22. При решении большинства подобных задач доступный объем данных превышает тот, который требуется для создания более простой языковой модели. В данной главе рассматриваются три конкретные задачи: информационный поиск (раздел 23.2), извлечение информации (раздел 23.3) и машинный перевод (раздел 23.4). Но вначале в ней представлен обзор вероятностных языковых моделей.

23.1. ВЕРОЯТНОСТНЫЕ ЯЗЫКОВЫЕ МОДЕЛИ

В главе 22 описана логическая модель языка; в ней для определения того, относится или не относится к некоторому языку данная строка, использовались грамматики CFG и DCG, а в данном разделе представлено несколько вероятностных моделей. Вероятностные модели имеют целый ряд преимуществ. Обучение этих моделей по имеющимся данным осуществляется очень просто: обучение сводится лишь к подсчету количества вариантов (с учетом определенных допусков на то, что из-за малого размера выборки могут возникать ошибки). Кроме того, эти модели являются более надежными (поскольку они способны принять любую строку, хотя и с низкой вероятностью); они отражают тот факт, что не все 100% говорящих на определенном языке согласны с тем, какие предложения фактически входят в состав языка; кроме того, такие модели могут использоваться для устранения неоднозначности, поскольку для выбора наиболее подходящей интерпретации могут применяться вероятностные законы.

✶ **Вероятностная языковая модель** позволяет определить распределение вероятностей множества строк (которое может быть бесконечно большим). К примерам таких моделей, которые уже рассматривались в данной книге, относятся двух- и трехсловные языковые модели (или модели двух- и трехсловных сочетаний), применявшиеся при распознавании речи (раздел 15.6). В однословной модели (или модели однословных сочетаний) каждому слову в словаре присваивается вероятность $P(w)$. В этой модели предполагается, что слова выбираются независимо, поэтому вероятность строки представляет собой произведение вероятностей входящих в нее слов и определяется выражением

$$\prod_i P(w_i).$$

Ниже приведена последовательность из 20 слов, которая была сформирована случайным образом из слов в оригинале данной книги с помощью однословной модели.

logical are as are confusion a may right tries agent goal the was diesel more object then information-gathering search is

В двухсловной модели каждому слову присваивается вероятность $P(w_i | w_{i-1})$ с учетом предыдущего слова. Часть данных о вероятностях таких двухсловных сочетаний приведена в табл. 15.2. Приведенная ниже случайная последовательность слов сформирована с помощью модели двухсловных сочетаний по материалам оригинала данной книги.

planning purely diagnostic expert systems are very similar computational approach would be represented compactly using tic tac toe a predicate

Вообще говоря, в модели n -словных сочетаний учитываются предыдущие $n-1$ слов и присваивается вероятность $P(w_i | w_{i-(n-1)} \dots w_{i-1})$. Приведенная ниже случайная последовательность сформирована с помощью модели трехсловных сочетаний по оригиналу данной книги.

planning and scheduling are integrated the success of naive bayes model is just a possible prior source by that time

Даже эти небольшие примеры позволяют понять, что модель трехсловных сочетаний превосходит модель двухсловных сочетаний (а последняя превосходит модель однословных сочетаний) как с точки зрения качества приближенного представления

текста на английском языке, так и с точки зрения успешной аппроксимации изложения темы в книге по искусственному интеллекту. Согласуются и сами модели: в модели трехсловных сочетаний строке, сформированной случайным образом, присваивается вероятность 10^{-10} , в модели двухсловных сочетаний — вероятность 10^{-29} , а в модели однословных сочетаний — вероятность 10^{-59} .

Но оригинал настоящей книги содержит всего лишь полмиллиона слов, поэтому в нем отсутствует достаточный объем данных для выработки качественной модели двухсловных сочетаний, не говоря уже о модели трехсловных сочетаний. Весь словарь оригинала данной книги включает примерно 15 тысяч различных слов, поэтому модель двухсловных сочетаний включает $15000^2 = 225$ миллионов пар слов. Безусловно, что вероятность появления по меньшей мере 99,8% этих пар будет равна нулю, но сама модель не должна указывать на то, что появление любой из этих пар в тексте невозможно. Поэтому требуется определенный способ **сглаживания** нулевых результатов фактического подсчета количества пар. Простейший способ выполнения этой задачи состоит в использовании так называемого способа **сглаживания с добавлением единицы**: к результатам подсчета количества всех возможных двухсловных сочетаний добавляется единица. Поэтому, если количество слов в текстовой совокупности равно N , а количество возможных двухсловных сочетаний равно B , то каждому двухсловному сочетанию с фактическим количеством c присваивается оценка вероятности $(c+1) / (N+B)$. Такой метод позволяет устранить проблему n -словных сочетаний с нулевой вероятностью, но само предположение, что все результаты подсчета количества должны быть увеличены точно на единицу, является сомнительным и может привести к получению некачественных оценок.

Еще один подход состоит в использовании метода **сглаживания с линейной интерполяцией**, в котором предусматривается объединение моделей трех-, двух- и однословных сочетаний с помощью линейной интерполяции. Оценка вероятности определяется по следующей формуле, с учетом того, что $c_3 + c_2 + c_1 = 1$:

$$\hat{P}(w_i | w_{i-2}w_{i-1}) = c_3 P(w_i | w_{i-2}w_{i-1}) + c_2 P(w_i | w_{i-1}) + c_1 P(w_i)$$

Параметры c_i могут быть заранее заданными или полученными путем обучения по алгоритму ЕМ. Существует возможность применения значений c_i , независимых от количества n -словных сочетаний, с тем, чтобы можно было присвоить больший вес оценкам вероятностей, полученным на основании больших значений количества.

Один из методов *оценки* языковой модели состоит в следующем. Вначале текстовая совокупность разделяется на обучающую совокупность и контрольную совокупность. Затем определяются параметры модели с помощью обучающих данных. После этого выполняется расчет вероятности, присвоенной контрольной совокупности с помощью данной модели; чем выше эта вероятность, тем лучше. Одним из недостатков этого подхода является то, что вероятность $P(words)$ при наличии длинных строк становится весьма небольшой; такие малые числовые значения могут вызвать антипереполнение в арифметике с плавающей точкой или просто стать неудобными для чтения. Поэтому вместо вероятности может быть вычислен **показатель связности** (perplexity) модели на контрольной строке слов *words* следующим образом:

$$Perplexity(words) = 2^{-\log_2(P(words)) / N}$$

где N — количество слов *words*. Чем ниже показатель связности, тем лучше модель. Модель n -словных сочетаний, которая присваивает каждому слову вероятность $1/k$,

имеет показатель связности k ; показатель связности может рассматриваться как средний коэффициент ветвления.

В качестве примера того, для чего может использоваться модель n -словных сочетаний, рассмотрим задачу ~~за~~ сегментации — поиска границ между словами в тексте без пробелов. Решением этой задачи обычно приходится заниматься при обработке текстов на японском и китайском языках, в которых отсутствуют пробелы между словами, но авторы полагают, что для большинства читателей более удобным будет пример из английского. Приведенное ниже предложение действительно несложно прочитать любому, кто знает английский язык.

Itiseasytoreadwordswithoutspaces

На первый взгляд может показаться, что для решения такой задачи приходится пользоваться всеми знаниями в области синтаксиса, семантики и прагматики английского языка. Но ниже будет показано, что в данном предложении можно легко восстановить пробелы с использованием простой модели однословных сочетаний.

В одной и предыдущих глав было показано, что для решения задачи поиска наиболее вероятной последовательности прохождения через решетку вариантов выбора слова может использоваться уравнение Витерби (15.9). А в листинге 23.1 приведен вариант алгоритма Витерби, специально предназначенный для решения задачи сегментации. Этот алгоритм принимает в качестве входных данных распределение вероятностей однословных сочетаний, $P(\text{word})$, и некоторую строку. Затем для каждой позиции i в данной строке этот алгоритм сохраняет в переменной $best[i]$ значение вероятности наиболее вероятной строки, которая охватывает участок от начала до позиции i . Кроме того, в этом алгоритме в переменной $words[i]$ сохраняется слово, оканчивающееся в позиции i , которое получило наибольшую вероятность. После того как по методу динамического программирования будут сформированы массивы $best$ и $words$, в алгоритме осуществляется обратный поиск через массив $words$ для определения наилучшего пути. В данном случае при использовании модели однословных сочетаний, соответствующей оригиналу этой книги, наиболее приемлемая последовательность слов действительно принимает вид “It is easy to read words without spaces” с вероятностью 10^{-25} . Сравнивая отдельные части этого предложения, можно обнаружить, что слово “easy” имеет вероятность однословного сочетания 2.6×10^{-4} , а альтернативный вариант его прочтения, “e as y”, имеет намного более низкую вероятность, 9.8×10^{-12} , несмотря на тот факт, что слова (точнее, имена переменных) “e” и “y” довольно часто встречаются в уравнениях данной книги. Аналогичным образом, другая часть этого предложения характеризуется следующими данными:

$P(\text{"without"}) = 0.0004$

$P(\text{"with"}) = 0.005$

$P(\text{"out"}) = 0.0008$

$P(\text{"with out"}) = 0.005 \times 0.0008 = 0.000004$

Листинг 23.1. Алгоритм сегментации строки на отдельные слова с помощью уравнения Витерби. Этот алгоритм восстанавливает наиболее вероятную сегментацию строки на слова после получения строки с удаленными пробелами

function Viterbi-Segmentation(*text*, *P*) **returns** последовательность наиболее подходящих слов *sequence* и значения вероятностей слов этой последовательности

```

inputs: text, строка символов с удаленными пробелами
         P, распределение вероятностей однословных сочетаний
         среди слов

n ← Length(text)
words ← пустой вектор длины n+1
best ← вектор длины n+1, первоначально полностью заполненный
        значениями 0.0
best[0] ← 1.0
/* Заполнить векторы best и words с помощью средств динамического
   программирования */
for i = 0 to n do
    for j = 0 to i-1 do
        word ← text[j:i]
        w ← Length(word)
        if P[word] × best[i - w] ≥ best[i] then
            best[i] ← P[word] × best[i - w]
            words[i] ← word
/* Теперь восстановить последовательность слов sequence */
sequence ← пустой список
i ← n
while i > 0 do
    продвинуть вектор words[i] в начало последовательности sequence
    i ← i - Length(words[i])
/* Последовательность наиболее подходящих слов, sequence, и
   значения вероятностей слов этой последовательности */
return sequence, best[i]

```

Поэтому слово “without” имеет в 100 раз более высокую вероятность, чем сочетание слов “with out”, согласно применяемой модели однословных сочетаний.

В данном разделе рассматривались модели *n*-элементных сочетаний, элементами которых являются слова, но широкое применение находят также модели *n*-элементных сочетаний, применяемые к другим элементам текста, таким как символы или части речи.

Вероятностные контекстно-свободные грамматики

В моделях *n*-элементных сочетаний используются статистические данные о совместном появлении элементов в текстовой совокупности, но эти модели не позволяют учитывать грамматические связи на расстояниях, превышающих *n*. В качестве альтернативной языковой модели может служить ~~э~~ **вероятностная контекстно-свободная грамматика**, или PCFG¹ (Probabilistic Context-Free Grammar), которая представляет собой такую грамматику CFG, где каждое правило подстановки имеет связанную с ним вероятность. Сумма вероятностей по всем правилам с одной и той же левой частью равна 1. Грамматика PCFG для части грамматики языка \mathcal{E}_0 приведена в листинге 23.2.

¹ Грамматики PCFG называют также *стохастическими контекстно-свободными грамматиками*, или SCFG (stochastic context-free grammar).

Листинг 23.2. Вероятностная контекстно-свободная грамматика (PCFG) и словарь для части грамматики языка \mathcal{E}_0 . Числа в квадратных скобках показывают вероятность того, что вместо символа в левой части правила будет выполнена подстановка правой части соответствующего правила

```

S → NP VP [1.00]
NP → Pronoun [0.10]
    | Name [0.10]
    | Noun [0.20]
    | Article Noun [0.50]
    | NP PP [0.10]
VP → Verb [0.60]
    | VP NP [0.20]
    | VP PP [0.20]
PP → Proposition NP [1.00]
Noun → breeze [0.10] | wumpus [0.15] | agent [0.05] | ...
Verb → sees [0.15] | smells [0.10] | goes [0.25] | ...
Pronoun → me [0.05] | you [0.10] | I [0.25] | it [0.20] | ...
Article → the [0.30] | a [0.35] | every [0.05] | ...
Proposition → to [0.30] | in [0.25] | on [0.05] | ...

```

В модели PCFG вероятность строки, $P(words)$, представляет собой сумму вероятностей деревьев синтаксического анализа этой строки. А вероятность данного конкретного дерева представляет собой произведение вероятностей всех правил, на основании которых сформированы узлы этого дерева. На рис. 23.1 показано, как вычислить вероятность некоторого предложения. Такую вероятность можно вычислить, применяя синтаксический анализатор диаграмм CFG для перечисления возможных вариантов синтаксического анализа, а затем складывая полученные вероятности. Но если нас интересует только наиболее вероятный вариант синтаксического анализа, то перебор всех маловероятных вариантов представляет собой бесполезную трату времени. Для эффективного поиска наиболее вероятного варианта синтаксического анализа может использоваться одна из разновидностей алгоритма Витерби или же какой-то метод поиска по первому наилучшему совпадению (такой как A*).

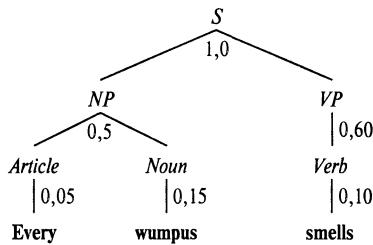


Рис. 23.1. Дерево синтаксического анализа для предложения “Every wumpus smells”, в котором показаны вероятности каждого поддерева. Вероятность всего дерева в целом равна $1.0 \times 0.5 \times 0.05 \times 0.15 \times 0.60 \times 0.10 = 0.000225$. Поскольку это дерево является единственным вариантом синтаксического анализа данного предложения, указанное число представляет собой также вероятность этого предложения

Недостатком грамматик PCFG является то, что они — контекстно-свободные. Это означает, что различие между $P(\text{"eat a banana"})$, “съешь банан”, и $P(\text{"eat a bandanna"})$, “съешь цветной платок”, зависит только от соотношения вероятностей $P(\text{"banana"})$ и $P(\text{"bandanna"})$, а не от вероятностей возникновения отношений между глаголом “eat” и соответствующими объектами. Для того чтобы можно было учитывать связи такого рода, нам потребуется контекстно-зависимая модель определенного типа наподобие **лексикализованной грамматики PCFG**, в которой определенную роль в оценке вероятности соответствующего словосочетания может играть голова² этого словосочетания. При наличии достаточного объема обучающих данных может быть получено правило для $VP \rightarrow VP\ NP$, обусловленное наличием головы входящего в него словосочетания VP (“eat”) и головы словосочетания NP (“banana”). Таким образом, лексикализованные грамматики PCFG позволяют учитывать некоторые ограничения на совместное вхождение элементов в моделях n -элементных сочетаний, наряду с грамматическими ограничениями моделей CFG.

Еще один недостаток состоит в том, что грамматики PCFG обнаруживают слишком заметное предпочтение по отношению к более коротким предложениям. В такой таковой совокупности, как архив журнала *Wall Street Journal*, средняя длина предложения составляет около 25 слов. Но обычно грамматика PCFG в конечном итоге присваивает гораздо более высокую вероятность таким правилам, как $S \rightarrow NP\ VP$, $NP \rightarrow Pronoun$ и $VP \rightarrow Verb$. Это означает, что грамматика PCFG присваивает весьма высокую вероятность многим коротким предложениям, таким как “He slept” (Он спал), тогда как в указанном журнале с большей вероятностью встречаются предложения наподобие следующего: “It has been reported by a reliable government source that the allegation that he slept is credible” (Из надежного правительственного источника поступило сообщение, согласно которому заявление о том, что он спал, заслуживает доверия). Создается впечатление, что словосочетания в этом журнале не являются контекстно-свободными; вместо этого его авторы оценивают допустимую ожидаемую длину предложения и используют полученную оценку в качестве мягкого глобального ограничения на структуру составляемых ими предложений. Такой подход к написанию текста трудно отразить в грамматике PCFG.

Определение с помощью обучения вероятностей для грамматики PCFG

При создании любой грамматики PCFG приходится преодолевать все сложности, связанные с формированием грамматики CFG, и наряду с этим решать проблему задания вероятностей для каждого правила. Такие обстоятельства наводят на мысль, что может оказаться более приемлемым подход, предусматривающий определение грамматики по имеющимся данным с помощью **обучения**, чем подход, основанный на инженерии знаний. Как и в случае распознавания речи, могут применяться данные двух типов — прошедшие и не прошедшие синтаксический анализ. Задача намного упрощается, если данные уже преобразованы в деревья с помощью синтаксического анализа лингвистами (или по меньшей мере носителями соответствующего естественного языка, прошедшими специальное обучение). Создание подобной тек-

² Головой словосочетания называется самое важное слово, например существительное в именном словосочетании.

стовой совокупности требует больших капиталовложений, и в настоящее время самые крупные из таких совокупностей содержат “всего лишь” около миллиона слов. А если имеется некоторая совокупность деревьев, то появляется возможность создать грамматику PCFG путем подсчета (и сглаживания). Для этого достаточно просмотреть все узлы, в которых корневым является каждый нетерминальный символ, и создать правило для каждой отдельной комбинации дочерних элементов в этих узлах. Например, если какой-то символ NP появляется 100 тысяч раз и имеется 20 тысяч экземпляров NP со списком дочерних элементов $[NP, PP]$, то создается правило

$$NP \rightarrow NP PP [0.20]$$

Если же текст не подвергнут синтаксическому анализу, то задача значительно усложняется. Это прежде всего связано с тем, что фактически приходится сталкиваться с двумя разными проблемами — определение с помощью обучения структуры грамматических правил и определение с помощью обучения вероятностей, связанных с каждым правилом (с аналогичным различием приходится сталкиваться при определении с помощью обучения параметров нейронных или байесовских сетей).

На данный момент примем предположение, что структура правил известна и предпринимается лишь попытка определить вероятности с помощью обучения. Для этого может использоваться подход на основе алгоритма ожидания—максимизации (expectation—maximization — EM), как и при обучении моделей НММ. В процессе обучения мы будем пытаться определить такие параметры, как вероятности правил. Скрытыми переменными являются деревья синтаксического анализа, поскольку неизвестно, действительно ли строка слов $w_1 \dots w_j$ сформирована с помощью правила $X \rightarrow \alpha$. На этапе E оценивается вероятность того, что каждая подпоследовательность сформирована с помощью каждого отдельного правила. Затем на этапе M оценивается вероятность каждого правила. Весь этот процесс вычисления может осуществляться в режиме динамического программирования с помощью алгоритма, называемого **внутренним—внешним алгоритмом**, по аналогии с прямым—обратным алгоритмом, применяемым для моделей НММ.

На первый взгляд причины продуктивного функционирования внутреннего—внешнего алгоритма кажутся непостижимыми, поскольку он позволяет успешно сформировать логическим путем грамматику на основании текста, не прошедшего синтаксический анализ. Но этот алгоритм имеет несколько недостатков. Во-первых, он действует медленно; этот алгоритм характеризуется временными затратами $O(n^3 t^3)$, где n — количество слов в предложении; t — количество нетерминальных символов. Во-вторых, пространство вероятностных присваиваний очень велико и практика показала, что при использовании этого алгоритма приходится сталкиваться с серьезной проблемой, связанной с тем, что он не выходит из локальных максимумов. Вместо него могут быть опробованы такие альтернативные варианты, как эмуляция отжига, за счет еще большего увеличения объема вычислений. В-третьих, варианты синтаксического анализа, присвоенные с помощью полученных в результате грамматик, часто трудно понять, а лингвисты находят их неудовлетворительными. В результате этого задача комбинирования знаний, представленных с помощью способов, приемлемых для человека, с данными, которые получены с помощью автоматизированного индуктивного логического вывода, становится затруднительной.

Определение с помощью обучения структуры правил для грамматики PCFG

Теперь предположим, что структура грамматических правил неизвестна. В таком случае сразу же возникает проблема, связанная с тем, что пространство возможных множеств правил является бесконечным, поэтому неизвестно, какое количество правил необходимо предусмотреть и какую длину должно иметь каждое правило. Один из способов решения этой проблемы состоит в том, чтобы организовать составление грамматики с помощью обучения в **нормальной форме Хомского**; это означает, что каждое правило должно находиться в одной из следующих двух форм:

$$X \rightarrow Y Z$$

$$X \rightarrow t$$

где X , Y и Z — нетерминальные символы; t — терминальный символ. В виде грамматики в нормальной форме Хомского, которая распознает точно такой же язык, может быть представлена любая контекстно-свободная грамматика. В таком случае появляется возможность принять произвольное ограничение, согласно которому количество нетерминальных символов будет равно n , и тем самым будет получено $n^3 + nv$ правил, где v — количество терминальных символов. Но практика показала, что такой подход является эффективным только применительно к небольшим грамматикам. Предложен также альтернативный подход, называемый **слиянием байесовских моделей**, аналогичный подходу с применением модели Sequitur (раздел 22.8). В этом подходе предусматривается формирование на первом этапе локальных моделей (грамматик) для каждого предложения, а затем использование минимальной длины описания для слияния моделей.

23.2. ИНФОРМАЦИОННЫЙ ПОИСК

Информационный поиск — это задача поиска документов, отвечающих потребностям пользователя в информации. Наиболее широко известными примерами систем информационного поиска являются поисковые машины World Wide Web. Пользователь Web может ввести в приглашении поисковой машины такой запрос, как [AI book], и получить список подходящих страниц. В данном разделе показано, как создаются подобные системы. Для систем информационного поиска (называемых сокращенно системами ИП) применяются перечисленные ниже характеристики.

1. Определение коллекции документов. В каждой системе должно быть принято определенное решение о том, что рассматривается в ней как документ — отдельный абзац, страница или многостраничный текст.
2. Способ формулировки **запроса** на **языке запросов**. Запрос указывает, какая информация требуется пользователю. Язык запросов может предусматривать лишь возможность составления списка слов, такого как [AI book], или может позволять задавать сочетание слов, которые должны быть расположены близко друг от друга, как в запросе ["AI book"]; он может содержать логические операторы, как в запросе [AI AND book]; а также включать операторы, отличные от логических, как в запросе [AI NEAR book] или [AI book SITE:www.aaai.org].

3. **Результирующий набор.** Таковым является подмножество документов, которые система информационного поиска определяет как **релевантные** данному запросу. Под словом *релевантный* подразумевается вероятно полезный (согласно конкретным информационным потребностям, сформулированным в запросе) для того лица, которое сформулировало запрос.
4. **Способ представления результирующего набора.** Он может быть настолько простым, как ранжированный список названий документов, или настолько сложным, как вращающаяся цветная карта результирующего набора, спроектированная на трехмерное пространство.

После чтения предыдущей главы могло сложиться впечатление, что систему информационного поиска возможно создать, преобразовав с помощью синтаксического анализа коллекцию документов в базу знаний, состоящую из логических высказываний, после чего в ней будет выполняться синтаксический анализ каждого запроса и поиск ответа в базе знаний с помощью предиката Ask. Но, к сожалению, еще никому не удалось создать крупномасштабную систему информационного поиска таким образом. Дело в том, что составить словарь и грамматику, которые охватывают большую коллекцию документов, слишком сложно, поэтому во всех системах информационного поиска используются более простые языковые модели.

Самые ранние системы информационного поиска действовали на основе **булевой модели ключевых слов**. Каждое слово в коллекции документов рассматривалось как булева характеристика, которая является истинной применительно к данному документу, если соответствующее слово встречается в документе, и ложной в противном случае. Поэтому характеристика “поиск” является истинной для текущей главы, но ложной для главы 15. В таком случае язык запросов представляет собой язык булевых выражений, заданных на характеристиках. Документ считается релевантным, только если соответствующее выражение принимает истинное значение. Например, запрос [информация AND поиск] принимает истинное значение для текущей главы и ложное для главы 15.

Преимуществом такой модели является то, что ее несложно описать и реализовать. Но она имеет некоторые недостатки. Во-первых, степень релевантности документа измеряется одним битом, поэтому отсутствуют руководящие данные, на основании которых можно было бы упорядочить релевантные документы для презентации. Во-вторых, булевы выражения могут оказаться непривычными для пользователей, не являющихся программистами или логиками. В-третьих, задача формулировки подходящего запроса может оказаться сложной даже для квалифицированного пользователя. Предположим, что предпринимается попытка выполнить запрос [информация AND поиск AND модели AND оптимизация], что приводит к получению пустого результирующего набора. После этого осуществляется попытка выполнить запрос [информация OR поиск OR модели OR оптимизация], но если он возвращает слишком большой объем результатов, то нелегко определить, какую попытку следует предпринять после этого.

В большинстве систем информационного поиска используются модели, основанные на статистических сведениях о количестве слов (а иногда и другие характеристики низкого уровня). В этой главе будет описана вероятностная инфраструктура, которая хорошо согласуется с описанными ранее языковыми моделями. Основная идея состоит в том, что после формулировки некоторого запроса требуется

найти документы, которые с наибольшей вероятностью будут релевантными по отношению к нему. Иными словами, необходимо вычислить следующее значение вероятности:

$$P(R=\text{true}|D, Q)$$

где D — документ; Q — запрос; R — булева случайная переменная, обозначающая релевантность. После получения этого значения можно применить принцип ранжирования вероятностей, который указывает, что если результирующий набор должен быть представлен в виде упорядоченного списка, это следует сделать в порядке уменьшения вероятности релевантности.

Существует несколько способов декомпозиции совместного распределения $P(R=\text{true}|D, Q)$. В настоящей главе будет описан подход, известный под названием **языкового моделирования**, в котором предусматривается получение оценки языковой модели для каждого документа, а затем вычисление для каждого запроса вероятности этого запроса с учетом языковой модели документа. Используя r для обозначения выражения $R=\text{true}$, можно переписать приведенное выше определение вероятности следующим образом:

$$\begin{aligned} P(r|D, Q) &= P(D, Q|r) P(r) / P(D, Q) && \text{(согласно правилу Байеса)} \\ &= P(Q|D, r) P(D|r) P(r) / P(D, Q) && \text{(согласно цепному правилу)} \\ &= \alpha P(Q|D, r) P(r|D) / P(D, Q) && \text{(согласно правилу Байеса,} \\ &&& \text{для фиксированного } D) \end{aligned}$$

Как уже было сказано, может быть предпринята попытка максимизировать значение $P(r|D, Q)$, но равным образом можно максимизировать отношение вероятностей $P(r|D, Q) / P(\neg r|D, Q)$. Это означает, что ранжирование документов может осуществляться на основе следующей оценки:

$$\frac{P(r|D, Q)}{P(\neg r|D, Q)} = \frac{P(Q|D, r) P(r|D)}{P(Q|D, \neg r) P(\neg r|D)}$$

Преимущество такого подхода состоит в том, что из процедуры вычисления устранился терм $P(D, Q)$. Теперь примем предположение, что в случае нерелевантных документов каждый документ является независимым по отношению к запросу. Иными словами, если какой-то документ нерелевантен по отношению к запросу, то получение информации о существовании этого документа не позволит определить, в чем состоит сам запрос. Это предположение может быть выражено с помощью такой формулы:

$$P(D, Q|\neg r) = P(D|\neg r) P(Q|\neg r)$$

На основании этого предположения получим следующее:

$$\frac{P(r|D, Q)}{P(\neg r|D, Q)} = P(Q|D, r) \times \frac{P(r|D)}{P(\neg r|D)}$$

Коэффициент $P(r|D) / P(\neg r|D)$ измеряет независимую от запроса вероятность того, что документ является релевантным. Таким образом, этот коэффициент представляет собой меру качества документа; некоторые документы с большей вероятностью будут релевантными по отношению к любому запросу, поскольку сами эти документы имеют изначально высокое качество. Применительно к статьям для академических журналов качество можно оценить на основании количества упоминаний об этих статьях в других источниках, а для оценки Web-страниц можно использовать

количество гиперссылок на ту или иную страницу. В каждом из этих случаев можно присвоить больший вес адресатам ссылок, характеризующимся высоким качеством. Одним из факторов оценки релевантности документа, независимой от запроса, может также служить продолжительность существования этого документа.

Первый коэффициент, $P(Q|D, r)$, представляет собой вероятность запроса с учетом релевантного документа. Для оценки этой вероятности необходимо выбрать языковую модель, описывающую то, как связаны запросы с релевантными документами. Один из широко распространенных подходов состоит в том, что документы представляются с помощью модели однословных сочетаний. В проблематике информационного поиска она известна также под названием модели **мультимножества слов**, поскольку в ней учитывается только частота появления каждого слова в документе, а не их порядок. При использовании такой модели следующие (очень короткие) примеры документов рассматриваются как идентичные: “man bites dog” (человек кусает собаку) и “dog bites man” (собака кусает человека). Очевидно, что эти документы имеют разный смысл, но верно также то, что оба они являются релевантными по отношению к запросам о собаках и укусах. Теперь, чтобы рассчитать вероятность запроса при наличии релевантного документа, достаточно просто перемножить вероятности слов в запросе, руководствуясь моделью однословных сочетаний данного документа. В этом и состоит **наивная байесовская** модель данного запроса. Используя Q_j для обозначения j -го слова в запросе, получим следующее:

$$P(Q|D, r) = \prod_j P(Q_j|D, r)$$

Это соотношение позволяет ввести такое упрощение:

$$\frac{P(r|D, Q)}{P(\neg r|D, Q)} = \prod_j P(Q_j|D, r) \frac{P(r|D)}{P(\neg r|D)}$$

Наконец, мы получили возможность применить эти математические модели к некоторому примеру. В табл. 23.1 приведены статистические данные по количеству однословных сочетаний применительно к словам в запросе [Bayes information retrieval model], выполняемом на коллекции документов, состоящей из пяти отдельных глав оригинала настоящей книги. Предполагается, что эти главы имеют одинаковое качество, поэтому требуется лишь вычислить вероятность запроса применительно к данному документу, для каждого документа. Такая процедура выполнена дважды, причем в первый раз использовалось выражение оценки несглаженного максимального правдоподобия D_1 , а во второй раз — модель D_1' со сглаживанием путем добавления единицы. Можно было бы предположить, что текущая глава должна получить наивысший ранг применительно к этому запросу, и в действительности были получены такие данные при использовании в каждой модели.

Преимуществом сглаженной модели является то, что она менее восприимчива к шуму и позволяет присвоить ненулевую вероятность релевантности документу, не содержащему все слова запроса. А преимуществом несглаженной модели является то, что она позволяет проще выполнить вычисления применительно к коллекциям с многочисленными документами, поскольку после создания индекса, где указано, в каких документах упоминается каждое слово, появляется возможность быстро формировать результирующий набор путем применения операции пересечения

к этим спискам, после чего остается вычислить $P(Q|D_i)$ только для документов, входящих в полученное пересечение, а не для каждого документа.

Таблица 23.1. Вероятностная модель информационного поиска для запроса [Bayes information retrieval model], применяемого к коллекции документов, состоящей из пяти глав оригинала настоящей книги. В этой таблице указано количество слов, относящееся к каждой паре “документ–слово”, и общее количество слов N для каждого документа. Используются две модели документа (D_i — это неслглаженная модель однословных сочетаний для i -го документа; D_i' — та же модель со сглаживанием путем добавления единицы) и вычисляется вероятность запроса применительно к каждому документу для обеих моделей. Очевидно, что текущая глава (глава 23) имеет наивысшие показатели при использовании любой модели, поскольку в ней появление искомых слов имеет в 200 раз более высокую вероятность по сравнению с любой другой главой

| Слова | Запрос | Глава 1 | Глава 13 | Глава 15 | Глава 22 | Глава 23 |
|----------------|--------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|
| Bayes | 1 | 5 | 32 | 38 | 0 | 7 |
| information | 1 | 15 | 18 | 8 | 12 | 39 |
| retrieval | 1 | 1 | 1 | 0 | 0 | 17 |
| model | 1 | 9 | 7 | 160 | 9 | 63 |
| N | 4 | 14680 | 10941 | 18186 | 16397 | 12574 |
| $P(Q D_i, r)$ | | 1.5×10^{-14} | 2.8×10^{-13} | 0 | 0 | 1.2×10^{-11} |
| $P(Q D_i', r)$ | | 4.1×10^{-14} | 7.0×10^{-13} | 5.2×10^{-13} | 1.7×10^{-15} | 1.5×10^{-11} |

Сравнительный анализ систем информационного поиска

Важная проблема состоит в том, как оценить показатели работы рассматриваемой системы информационного поиска. Проведем эксперимент, в котором системе предъявляется ряд запросов, а результирующие наборы оцениваются с учетом суждений людей о релевантности полученных результатов. По традиции при такой оценке применяются два критерия: полнота выборки и точность. Сформулируем определения этих критериев с помощью примера. Предположим, что некоторая система информационного поиска возвратила результирующий набор, относящийся к одному запросу, применительно к которому известно, какие документы являются и не являются релевантными, из совокупности в 100 документов. Количество документов в каждой категории приведено в табл. 23.2.

Таблица 23.2. Количество документов в каждой категории

| | В результирующем наборе | Не в результирующем наборе |
|----------------|-------------------------|----------------------------|
| Релевантный | 30 | 20 |
| Не релевантный | 10 | 40 |

Показатель \approx **точности** измеряет долю документов в результирующем наборе, которые действительно являются релевантными. В данном примере точность составляет $30/(30+10)=0,75$. Относительное количество ложных положительных оценок равно $1-0,75=0,25$. Показатель \approx **полноты выборки** измеряет долю всех релевантных документов в коллекции, которые находятся в результирующем наборе. В данном примере полнота выборки составляет $30/(30+20)=0,60$. Относительное количество ложных отрицательных оценок равно $1-0,60=0,40$. Вычисление показателя полноты выборки в очень большой коллекции документов, такой как World Wide Web, стано-

вится сложным, поскольку отсутствует удобный способ проверки каждой страницы в Web на релевантность. Самое лучшее решение, которое может быть принято в данном случае, состоит в том, чтобы оценивать полноту выборки путем исследования определенной части документов или совсем игнорировать показатель полноты выборки и оценивать коллекцию документов только по показателю точности.

В некоторых системах может происходить потеря точности из-за увеличения полноты выборки. В крайнем случае в системе, которая возвращает в составе результирующего набора каждый документ из коллекции документов, гарантированно достигается полнота выборки, равная 100%, но точность становится низкой. Еще один вариант состоит в том, что система может возвращать единственный документ и показывать низкую полноту выборки, но достигать высокой вероятности получения 100%-ной точности. Один из способов достижения компромисса между точностью и полнотой выборки состоит в использовании \simeq **кривой ROC**. Аббревиатура “ROC” сокращенно обозначает показатель “рабочая характеристика приемника” (receiver operating characteristic), который требует дополнительных пояснений. Он представляет собой график, на котором относительное количество ложных отрицательных оценок измеряется по оси y , а относительное количество ложно положительных оценок измеряется по оси x , что позволяет находить различные точки компромиссов. Площадь под этой кривой представляет собой суммарную оценку эффективности системы информационного поиска.

Показатели полноты выборки и точности были определены в то время, когда задачи информационного поиска решались главным образом библиотекарями, которые были заинтересованы в получении исчерпывающих, научно обоснованных результатов. В настоящее время большинство запросов (количество которых измеряется сотнями миллионов в сутки) выполняется пользователями Internet, которых в меньшей степени интересует исчерпывающая полнота ответов и требуется лишь немедленно найти ответ. Для таких пользователей одним из наиболее приемлемых критериев является средний \simeq **обратный ранг** первого релевантного результата. Это означает, что если первый результат, полученный системой, является релевантным, он получает применительно к данному запросу оценку 1, а если первые два результата не релевантны, а третий является таковым, он получает оценку 1/3. Еще одним критерием служит \simeq **время ожидания ответа**, который позволяет измерить продолжительность времени, требуемую для поиска желаемого ответа на поставленный пользователем вопрос. Этот показатель лучше оценивает те характеристики систем информационного поиска, которые действительно хотелось бы точно измерить, но обладает одним недостатком, связанным с тем, что для проведения каждого нового эксперимента приходится привлекать новую партию испытуемых субъектов — людей.

Совершенствование информационного поиска

В модели однословных сочетаний все слова рассматриваются как полностью независимые, но носителям языка известно, что некоторые слова обладают определенными связями, например, слово “couch” (кушетка) тесно связано со словами “couches” и “sofa”. Во многих системах информационного поиска предпринимаются попытки учитывать подобные корреляции.

Например, если запрос сформулирован как [couch], то исключение из результирующего набора таких документов, в которых упоминаются слова “COUCH” или

“couches”, но не “couch”, было бы неправильным. В большинстве систем информационного поиска используются средства **приведения к нижнему регистру**, с помощью которых слово “COUCH” преобразуется в “couch”, а во многих дополнительно применяется алгоритм **выделения основы**, позволяющий преобразовать слово “couches” в основную форму “couch”. Применение указанных средств обычно позволяет добиться небольшого увеличения полноты выборки (для английского языка такое увеличение составляет порядка 2%). Но использование таких средств может привести к снижению точности. Например, после преобразования слова “stocking” в “stock” с помощью выделения основы обычно снижается точность применительно к запросам, относящимся либо к чулочно-носочным изделиям, либо к финансовым инструментам, хотя и может увеличить полноту выборки применительно к запросам о ведении домашнего хозяйства. Алгоритмы выделения основы, действующие с помощью фиксированных правил (например, правил, предусматривающих удаление суффикса “-ing”), не позволяют предотвратить возникновение этой проблемы, но новейшие алгоритмы, действующие на базе словаря (в которых суффикс “-ing” не удаляется, если слово с этим суффиксом имеется в словаре), позволяют решить эту проблему. Применение средств выделения основы в английском языке не позволяет добиться существенных результатов, но играет более важную роль в других языках. Например, в тексте на немецком языке нередко можно встретить слова наподобие “Lebensversicherungsgesellschaftsangestellter” (служащий компании страхования жизни). В таких языках, как финский, турецкий, инупик и юпик, имеются рекурсивные морфологические правила, которые позволяют в принципе составлять слова неограниченной длины.

Следующий этап состоит в том, что в системе предусматривается распознавание **синонимов**, например, таких, как “sofa” и “couch”. Как и при использовании средств выделения основы, это позволяет добиться небольшого увеличения полноты выборки, но при непродуманном использовании этих средств возникает опасность снижения точности. Пользователи, желающие получить информацию о футболисте Тиме Коуче (Tim Couch), вряд ли хотели бы погрузиться в бесконечные объемы сведений о кушетках и диванах. Проблема состоит в том, что “языки не терпят абсолютной синонимии, так же как природа не терпит вакуума” [312]. Это означает, что при появлении в языке двух слов, соответствующих одному и тому же понятию, люди, говорящие на этом языке, совместными усилиями уточняют толкование таких слов для устранения путаницы.

Во многих системах информационного поиска в определенной степени используются **двухсловные сочетания**, но полная вероятностная модель двухсловных сочетаний реализована лишь в немногих системах. Кроме того, для исправления опечаток как в документах, так и в запросах могут использоваться процедуры **коррекции орфографических ошибок**.

В качестве последнего усовершенствования можно указать, что повышение качества функционирования системы информационного поиска достигается также с помощью использования **метаданных** — данных, внешних по отношению к тексту самого документа. К примерам таких данных относятся ключевые слова, подготовленные разработчиком документа, и гипертекстовые ссылки между документами.

Способы представления результирующих наборов

В соответствии с принципом вероятностного ранжирования должен быть получен результирующий набор и представлен пользователю в виде списка, отсортированного с учетом вероятности релевантности. Такой способ представления имеет смысл, если пользователь заинтересован в поиске всех релевантных документов, проведенном настолько быстро, насколько это возможно. Но он оказывается не совсем приемлемым, поскольку в нем не учитывается полезность. Например, если в коллекции имеются две копии наиболее релевантного документа, то после просмотра первой копии полезность второй, имеющей такую же релевантность, становится равной нулю. Во многих системах информационного поиска имеются механизмы, позволяющие исключать результаты, которые слишком подобны ранее полученным результатам.

Один из наиболее мощных способов повышения производительности системы информационного поиска состоит в обеспечении возможности использовать **отзывы, касающиеся релевантности**. В этих отзывах пользователь указывает, какие документы из первоначального результирующего набора являются релевантными. После этого система может представить второй результирующий набор документов с документами, подобными указанным.

Еще один подход состоит в том, что результирующий набор представляется в виде размеченного дерева, а не упорядоченного списка. С помощью средств **классификации документов** эти результаты оформляются в виде заранее определенной таксономии тем. Например, коллекция новостных сообщений может классифицироваться на “World News” (Зарубежные новости), “Local News” (Местные новости), “Business” (Новости экономики), “Entertainment” (Новости культуры) и “Sports” (Новости спорта). А при использовании средств **кластеризации документов** дерево категорий создается для каждого результирующего набора с нуля. Методы классификации являются приемлемыми, если количество тем в коллекции невелико, а методы кластеризации в большей степени подходят для более широких коллекций, таких как World Wide Web. И в том и в другом случае после выполнения запроса пользователя результирующий набор предъявляется ему в виде папок, составленных по категориям.

Классификация — это задача контролируемого обучения, поэтому для ее решения может применяться любой из методов, описанных в главе 18. Один из широко используемых подходов состоит в формировании деревьев решений. После подготовки обучающего множества документов, обозначенных правильными категориями, может быть сформировано единственное дерево решений, листьям которого поставлены в соответствие документы, принадлежащие к той или иной категории. Такой подход полностью себя оправдывает, если имеется лишь несколько категорий, но при наличии более крупных множеств категорий приходится формировать по одному дереву решений для каждой категории, притом что листья этого дерева обозначают документ как принадлежащий или не принадлежащий к данной категории. Обычно характеристиками, проверяемыми в каждом узле, являются отдельные слова. Например, в одном из узлов дерева решений для категории “Sports” может быть предусмотрена проверка наличия слова “basketball”. Для классификации текстов были опробованы такие средства, как усиленные деревья решений, наивные байесовские модели и машины поддерживающих векторов; во многих случаях точность при использовании булевой классификации находилась в пределах 90–98%.

Кластеризация относится к типу задач неконтролируемого обучения. В разделе 20.3 было показано, как может использоваться алгоритм ЕМ для улучшения начальной оценки кластеризации на основе сочетания гауссовых моделей. Задача кластеризации документов является более сложной, поскольку неизвестно, было ли выполнено формирование данных с помощью правильной гауссовой модели, а также в связи с тем, что приходится действовать в условиях пространства поиска, имеющего намного больше размерностей. Для решения этой задачи был разработан целый ряд подходов.

В методе **агломеративной кластеризации** создается дерево кластеров путем выполнения полной обработки совокупности вплоть до отдельных документов. Отсечение ветвей этого дерева для получения меньшего количества категорий может быть выполнено на любом уровне, но такая операция рассматривается как выходящая за рамки самого алгоритма. На первом этапе каждый документ рассматривается как отдельный кластер. После этого отыскиваются два кластера, наиболее близкие друг к другу согласно определенному критерию расстояния, и эти два кластера сливаются в один. Такой процесс повторяется до тех пор, пока не остается только один кластер. Критерием расстояния между двумя документами является некоторый критерий, измеряющий совпадение слов в этих документах. Например, документ может быть представлен как вектор количества слов, а само расстояние определено как евклидово расстояние между двумя векторами. Критерием расстояния между двумя кластерами может служить расстояние до середины кластера или может учитываться среднее расстояние между элементами кластеров. Метод агломеративной кластеризации требует затрат времени, пропорциональных $O(n^2)$, где n — количество документов.

В методе **кластеризации по k средним** создается плоское множество, состоящее точно из k категорий. Этот метод действует, как описано ниже.

1. Случайным образом осуществляется выборка k документов для представления k категорий.
2. Каждый документ обозначается как принадлежащий к ближайшей категории.
3. Вычисляется среднее каждого кластера и используются k средних для представления новых значений k категорий.
4. Этапы 2) и 3) повторяются до тех пор, пока алгоритм не сходится.

Для метода кластеризации по k средним требуются затраты времени, пропорциональные $O(n)$, в чем состоит одно из его преимуществ над агломеративной кластеризацией. Но в литературе часто приходится встречать сообщение о том, что этот метод является менее точным по сравнению с агломеративной кластеризацией, хотя некоторые исследователи сообщают, что он позволяет добиться почти таких же высоких показателей [1460].

Но независимо от применяемого алгоритма кластеризации требуется решить еще одну задачу, прежде чем результаты кластеризации можно будет использовать для представления результирующего набора, — найти удобный способ описания кластера. При использовании метода классификации имена категорий уже определены (например, “Earnings” — доходы), но при кластеризации имена категорий приходится изобретать заново. Один из способов выполнения этой задачи состоит в подборе списка слов, которые являются представительными для этого кластера. Еще один вариант состоит в применении названий одного или нескольких документов, близких к центру кластера.

Создание систем информационного поиска

До сих пор в этой главе было приведено описание работы систем информационного поиска в общих чертах, но не показано, как добиться эффективного функционирования этих систем для того, чтобы машины поиска Web могли возвращать искомые результаты обработки коллекции, состоящей из многих миллиардов страниц, за десятые доли секунды. Двумя основными структурами данных любой системы информационного поиска являются лексикон, содержащий списки всех слов в рассматриваемой коллекции документов, и инвертированный индекс, в котором перечислены все места, где каждое слово встречается в коллекции документов.

Лексиконом называется структура данных, которая поддерживает одну важную операцию: после получения определенного слова она возвращает данные о том, в каком месте инвертированного индекса хранятся экземпляры этого слова. В некоторых версиях систем информационного поиска эта структура возвращает также данные об общем количестве документов, содержащих искомое слово. Лексикон должен быть реализован с использованием хэш-таблицы или аналогичной структуры данных, которая обеспечивает быстрое выполнение этой операции поиска. Иногда в лексикон не включают ряд широко распространенных слов, имеющих малое информационное содержание. Эти слова, называемыми **запретными словами** (“the”, “of”, “to”, “be”, “a” и т.д.), только занимают место в индексе, но не увеличивают ценность результата. Единственным резонным основанием для включения их в лексикон может служить вариант, в котором лексикон используется для поддержки фразовых запросов, — индекс, содержащий запретные слова, необходим для эффективной выборки результатов для таких запросов, как “to be or not to be”.

Инвертированный индекс³, подобно индексу (предметному указателю), приведенному в конце данной книги, состоит из множества **списков позиций** — обозначений тех мест, где встречается каждое слово. Применительно к булевой модели ключевых слов список позиций представляет собой список документов. А список позиций, применяемый в модели однословных сочетаний, представляет собой список пар (документ, количество). Для обеспечения поддержки фразового поиска список позиций должен также включать обозначения позиций в каждом документе, где встречается каждое слово.

Если запрос состоит из одного слова (а такая ситуация встречается в 26% случаев, согласно [1411]), его обработка происходит очень быстро. Для этого выполняется единственная операция поиска в лексиконе для получения адреса списка позиций, а затем создается пустая очередь по приоритету. В дальнейшем происходит обработка списка позиций одновременно по одному документу и проверка количества экземпляров искомого слова в документе. Если очередь по приоритету имеет меньше, чем R элементов (где R — размер желаемого результирующего набора), то пара (документ, количество) добавляется к очереди. В противном случае, если количество экземпляров искомого слова больше по сравнению с соответствующими данными элемента с наименьшими показателями в очереди по приоритету, этот элемент

³ Термин “инвертированный индекс” (inverted index) является избыточным; лучшим термином был бы просто “индекс”. Индекс называют инвертированным потому, что он задает порядок расположения слов, отличный от того порядка, в котором слова расположены в тексте, но таковы все индексы. Тем не менее по традиции в системах информационного поиска применяется термин “инвертированный индекс”.

с наименьшими показателями удаляется из очереди и добавляется новая пара (документ, количество). Таким образом, поиск ответа на запрос требует затрат времени, пропорциональных $O(H+R\log R)$, где H — количество документов в списке позиций. Если запрос состоит из n слов, требуется выполнить слияние n списков позиций, для чего требуется затраты времени, равные $O(nH+R\log R)$.

В данной главе теоретический обзор средств информационного поиска представлен с использованием вероятностной модели, поскольку эта модель основана на идеях, уже описанных при изложении других тем в настоящей книге. Но в системах информационного поиска, фактически применяемых на практике, чаще всего используется другой подход, называемый **моделью векторного пространства**. Эта модель основана на таком же подходе с использованием мультимножества слов, как и вероятностная модель. Каждый документ представлен в виде вектора частот однословных сочетаний. Запрос также представляется полностью аналогичным образом; например, запрос [Bayes information retrieval model] представляется в виде вектора:

[0, ..., 1, 0, ..., 1, 0, ..., 1, 0, ..., 1, 0, ...]

Применяемая здесь идея состоит в том, что существует по одному измерению для каждого слова в коллекции документов, а запрос получает оценку 0 по каждому измерению, кроме тех четырех, которые фактически присутствуют в запросе. Релевантные документы отбираются путем поиска среди векторов документов именно тех векторов, которые являются ближайшими соседями по отношению к вектору запроса в векторном пространстве. Одним из критериев подобия служит точечное произведение между вектором запроса и вектором документа; чем больше это произведение, тем ближе два вектора. С точки зрения алгебры указанные вычисления обеспечивают получение высоких оценок теми словами, которые часто появляются и в документе, и в запросе. А с точки зрения геометрии точечное произведение между двумя векторами равно косинусу угла между этими векторами; максимизация косинуса угла между двумя векторами (находящимися в одном и том же квадранте) равносильна уменьшению этого угла до нуля.

Это краткое описание далеко не исчерпывает всю проблематику модели векторного пространства. На практике эта модель была развита до такой степени, чтобы в ней можно было учесть целый ряд дополнительных средств, уточнений, исправлений и дополнений. Основная идея ранжирования документов по их подобию в векторном пространстве позволяет внести новые понятия в систему числового ранжирования. Некоторые специалисты утверждают, что вероятностная модель позволила бы выполнять аналогичные манипуляции более научно обоснованным способом, но исследователи в области информационного поиска вряд ли согласятся перейти на другой инструментарий до тех пор, пока не убедятся в явных преимуществах другой модели с точки зрения производительности.

Для того чтобы получить представление о том, с какими масштабами применения средств индексации приходится сталкиваться при решении типичной задачи информационного поиска, рассмотрим стандартную совокупность документов из коллекции TREC (Text REtrieval Conference), состоящую из 750 тысяч документов с общим объемом в 2 Гбайт текста. Лексикон этой коллекции содержит приблизительно 500 тысяч слов, к которым применены операции выделения основы и приведения к нижнему регистру; для хранения этих слов требуется объем памяти от 7 до 10 Мбайт. Инвертированный индекс с парами (документ, количество) занимает 324

Мбайт, хотя и остается возможность применить методы сжатия для сокращения этого объема до 83 Мбайт. Методы сжатия позволяют экономить пространство за счет небольшого увеличения потребностей в обработке. Но если сжатие позволяет держать весь индекс в памяти, а не хранить его на диске, то появляется возможность добиться существенного общего прироста производительности. Для поддержки фразовых запросов требуется увеличение этого объема примерно до 1200 Мбайт не в сжатом виде или до 600 Мбайт со сжатием. Машины поиска Web действуют в масштабах, превышающих примерно в 3000 раз указанные выше. При этом многие из описанных здесь проблем остаются теми же, а поскольку задача оперирования с терабайтами данных в одном компьютере практически не осуществима, индекс разделяется на k сегментов и каждый сегмент сохраняется на отдельном компьютере. Запрос передается параллельно на все компьютеры, а затем k результирующих наборов сливаются в один результирующий набор, который предъявляется пользователю. Кроме того, машины поиска Web вынуждены справляться с тысячами запросов, поступающих в секунду, поэтому для них требуется n копий k компьютеров. Со временем значения k и n продолжают возрастать.

23.3. ИЗВЛЕЧЕНИЕ ИНФОРМАЦИИ

✎ **Извлечением информации** называется процесс создания записей базы данных путем просмотра текста и выявления экземпляров конкретного класса объектов или событий, а также связей между этими объектами и событиями. Может быть принята попытка применить такой процесс для извлечения данных об адресах из Web-страниц и внесения в базу данных информации об улице, городе, штате и почтовом коде или извлечения сведений о происходящих штормах из сообщений о погоде и внесения в базу данных информации о температуре, скорости ветра и количестве осадков. Системы извлечения информации занимают промежуточное положение между системами информационного поиска и полными синтаксическими анализаторами текста, поскольку к ним предъявляются более высокие требования, чем просто преобразование документа в мультимножество слов, но меньшие требования по сравнению с полным анализом каждого предложения.

Простейшим типом системы извлечения информации является система, основанная на атрибутах, поскольку в ней предполагается, что весь текст относится к одному объекту и задача состоит в извлечении атрибутов этого объекта. Например, в разделе 10.5 упоминалась задача извлечения из текста “17in SXGA Monitor for only \$249.99” отношений базы данных, определяемых следующим выражением:

$$\exists m \in \text{ComputerMonitors} \wedge \text{Size}(m, \text{Inches}(17)) \wedge \text{Price}(m, \$ (249.99)) \\ \wedge \text{Resolution}(m, 1280 \times 1024)$$

Определенная часть этой информации может обрабатываться с помощью ✎ **регулярных выражений**, которые определяют регулярную грамматику, заданную на одной строке текста. Регулярные выражения используются в командах Unix, таких как `grep`, в языках программирования, таких как Perl, и в текстовых процессорах, таких как Microsoft Word. Подробные сведения о грамматике, применяемой в том или ином инструментальном средстве, в значительной степени различаются, поэтому их лучше всего узнать из соответствующего справочного руководства, но в

табл. 23.3 показано, как сформировать регулярное выражение для выделения данных о ценах в долларах, и продемонстрировано применение общих подвыражений.

Таблица 23.3. Примеры применения регулярных выражений

| Регулярное выражение | Результат применения |
|------------------------|--|
| [0-9] | Согласуется с любой цифрой от 0 до 9 |
| [0-9]+ | Согласуется с одной или большим количеством цифр |
| .[0-9][0-9] | Согласуется с конструкцией, состоящей из точки, за которой следуют две цифры |
| (.[0-9][0-9])? | Согласуется с конструкцией, состоящей из точки, за которой следуют две цифры, или с пустой строкой |
| \$[0-9]+(.[0-9][0-9])? | Согласуется со строкой \$249.99, или \$1.23, или \$1000000, или ... |

Системы извлечения информации на основе атрибутов могут быть созданы в виде ряда регулярных выражений, по одному для каждого атрибута. Если регулярное выражение согласуется с текстом один и только один раз, то существует возможность извлечь часть текста, определяющую значение соответствующего атрибута. Если соответствия не найдены, то больше ничего нельзя сделать, а если регулярное выражение согласуется с текстом в нескольких местах, то нужно применить процесс осуществления выбора между этими согласованиями. Одна из возможных стратегий состоит в том, чтобы для каждого атрибута было предусмотрено несколько регулярных выражений, упорядоченных по приоритетам. Поэтому, например, регулярное выражение с наивысшим приоритетом для выделения цены может предусматривать применение строки “our price:”, за которой сразу же следует знак доллара “\$”; если же эта строка не будет обнаружена, можно сразу же перейти к использованию менее надежного регулярного выражения. Еще одна стратегия состоит в том, чтобы найти все согласования и применить определенный способ выбора между ними. Например, можно взять самую низкую цену, которая находится в пределах 50% от самой высокой цены. Это позволит обрабатывать тексты, подобные следующему: “List price \$99.00, special sale price \$78.00, shipping \$3.00”.

На более высоком этапе развития по сравнению с системами извлечения информации на основе атрибутов находятся системы извлечения информации на основе отношений, или реляционные системы, которые позволяют учитывать наличие в тексте информации о более чем одном объекте и отношениях между ними. Таким образом, при обнаружении такими системами текста “\$249.99” они должны определить не только цену, но и объект, имеющий эту цену. Типичной системой извлечения информации на основе отношений является система FASTUS, которая применяется для обработки новостных сообщений о корпоративных слияниях и приобретениях. Эта система способна прочитать следующее сообщение:

Bridgestone Sports Co. said Friday it has set up a joint venture in Taiwan with a local concern and a Japanese trading house to produce golf clubs to be shipped to Japan.

и сформировать примерно такую запись базы данных:

```
e ∈ JointVentures ∧ Product(e, "golf clubs") ∧ Date(e, "Friday")
  ∧ Entity(e, "Bridgestone Sports Co") ∧ Entity(e, "a local concern")
  ∧ Entity(e, "a Japanese trading house")
```

Реляционные системы извлечения информации часто создаются на основе ~~каскадных преобразователей с конечными автоматами~~. Это означает, что они состоят из ряда конечных автоматов (Finite-State Automaton — FSA), где каждый автомат принимает текст в качестве входных данных, преобразует этот текст в другой формат и передает его следующему автомату. Такой способ обработки является осуществимым, поскольку каждый конечный автомат действует достаточно эффективно, а при совместном использовании они приобретают способность извлекать необходимую информацию. Типичной системой такого типа является FASTUS, которая состоит из конечных автоматов, выполняющих описанные ниже пять этапов обработки.

1. Разбиение на лексемы.
2. Обработка сложных слов.
3. Обработка базовых групп.
4. Обработка сложных фраз.
5. Слияние структур.

Первым этапом обработки системы FASTUS является **разбиение на лексемы**, в котором поток символов сегментируется на лексемы (слова, числа и знаки препинания). Применительно к тексту на английском языке разбиение на лексемы может быть выполнено довольно просто; для этого достаточно лишь следить за разделяющими символами пробелами или знаками препинания. А применительно к тексту на японском языке для разбиения на лексемы требуется вначале выполнить сегментацию, используя нечто вроде алгоритма сегментации Витерби (см. листинг 23.1). Некоторые средства разбиения на лексемы позволяют также обрабатывать такие языки разметки, как HTML, SGML и XML.

На втором этапе обрабатываются **сложные слова**, включая такие словосочетания, как “set up” (настройка) и “joint venture” (совместное предприятие), а также имена собственные, такие как “Prime Minister Tony Blair” и “Bridgestone Sports Co.”. Сложные слова распознаются с использованием сочетания лексических элементов и грамматических правил конечного автомата. Например, название компании может быть распознано с помощью следующего правила:

```
CapitalizedWord+ ("Company" | "Co" | "Inc" | "Ltd")
```

Эти правила необходимо составлять с учетом всех предосторожностей и проверять на полноту и точность. Одна из коммерческих систем распознала словосочетание “Intel Chairman Andy Grove” (Председатель правления Intel Энди Гроув) как обозначение местности, а не имя лица, применив правило в следующей форме:

```
CapitalizedWord+ ("Grove" | "Forest" | "Village" | ...)
```

На третьем этапе выполняется обработка **базовых групп**; под этим подразумеваются именные и глагольные группы. Общая идея состоит в том, чтобы объединить на этом этапе слова в такие элементы, которые можно будет легко обрабатывать на последующих этапах. Именная группа состоит из заглавного существительного, за которым следуют необязательные определители и другие модификаторы. Поскольку именная группа не включает всех сложных конструкций, предусмотренных для именного словосочетания NP в грамматике \mathcal{E}_1 , не требуются рекурсивные правила контекстно-свободной грамматики — достаточно только использовать правила регу-

лярной грамматики, допустимые для конечных автоматов. Глагольная группа состоит из глагола и присоединенных к нему вспомогательных частиц и наречий, но без прямого и косвенного объекта и пропозициональных предложений. Предложение, приведенное выше в качестве примера, может быть преобразовано на этом этапе в следующую конструкцию:

| | | |
|-------------------------------|--|----------------------------------|
| 1. NG: Bridgestone Sports Co. | | 10. NG: a local concern |
| 2. VG: said | | 11. CJ: and |
| 3. NG: Friday | | 12. NG: a Japanese trading house |
| 4. NG: it | | 13. VG: to produce |
| 5. VG: had set up | | 14. NG: golf clubs |
| 6. NG: a joint venture | | 15. VG: to be shipped |
| 7. PR: in | | 16. PR: to |
| 8. NG: Taiwan | | 17. NG: Japan |
| 9. PR: with | | |

где NG обозначает именную группу; VG — глагольную группу; PR — предлог, CJ — союз.

На четвертом этапе базовые группы объединяются в **сложные фразы**. И в этом случае цель состоит в том, чтобы применяемые правила могли быть реализованы с помощью конечного автомата и допускали быструю обработку, а полученный результат сводился к непротиворечивым (или почти непротиворечивым) выходным фразам. В правиле комбинирования одного из типов учитываются события, типичные для рассматриваемой проблемной области. Например, следующее правило отражает один из способов описания процесса формирования совместного предприятия:

Company+SetUp JointVenture("with" Company+)?

Этот этап является первым из каскада этапов, в которых полученные выходные данные помещаются в шаблон базы данных, а также выводятся в выходной поток.

На последнем этапе происходит **слияние структур**, которые были сформированы на предыдущем этапе. Если в следующем предложении сказано: “The joint venture will start production in January” (Это совместное предприятие начнет выпускать продукцию в январе), то на данном этапе будет отмечено, что в двух ссылках на совместное предприятие (“joint venture”) упоминается один и тот же объект, и они будут объединены в одну ссылку.

Вообще говоря, средства извлечения информации действуют успешно применительно к ограниченной проблемной области, в которой возможно заранее определить, какие темы будут обсуждаться и в каких терминах будет проходить это обсуждение. Такие средства показали свою применимость для целого ряда проблемных областей, но они не способны заменить полномасштабную обработку текста на естественном языке.

23.4. МАШИННЫЙ ПЕРЕВОД

Машинным переводом называется автоматический перевод текста с одного естественного языка (исходного) на другой (целевой). Практика показала, что этот процесс может применяться для выполнения целого ряда задач, включая перечисленные ниже.

1. Грубый перевод, цель которого состоит в том, чтобы только определить смысл отрывка текста. В нем допускается наличие грамматически неправильных и неуклюжих предложений, при условии, что смысл этих предложений ясен. Например, в ходе Web-серфинга пользователю часто достаточно получить грубый перевод Web-страницы на иностранном языке. Иногда лицо, владеющее только родным языком, может успешно выполнить последующее редактирование результатов перевода без необходимости читать источник. Такого рода перевод с помощью машины позволяет сэкономить деньги, поскольку тем, кто занимается редактированием текста, полученного с помощью машинного перевода, можно платить меньше, чем тем, кто непосредственно переводит с иностранного языка.
2. Перевод источников с ограниченной тематикой, в котором тема и формат исходного текста строго ограничены. Одним из наиболее удачных примеров является система Taum-Meteo, которая переводит сообщения о погоде с английского языка на французский. Ее работа основана на том, что язык, используемый в сообщениях о погоде, является в высшей степени стилизованным и формализованным.
3. Перевод с предварительным редактированием, в котором исходный документ заранее редактируется перед машинным переводом людьми для того, чтобы он соответствовал ограниченному подмножеству английского или любого другого языка оригинала. Такой подход является особенно экономически эффективным, если есть необходимость перевести один документ на много языков, как в случае распространения юридических документов в Европейском экономическом сообществе или в случае тиражирования инструкций компаниями, которые продают один и тот же продукт во многие страны. Ограниченные языки иногда называют “Caterpillar English” (английским языком Caterpillar), поскольку впервые попытку оформлять свои инструкции в такой форме предприняла корпорация Caterpillar. Компания Хегох определила язык для своих инструкций по техническому обслуживанию, который является достаточно простым для того, чтобы его можно было перевести с помощью машины на языки всех стран, с которыми компания Хегох имеет деловые связи. Дополнительным преимуществом оказалось то, что оригинальные английские инструкции также стали более понятными.
4. Литературный перевод, в котором сохраняются все нюансы исходного текста. В настоящее время эта задача выходит за рамки возможностей машинного перевода.

В качестве примера грубого перевода в табл. 23.4 приведен первый абзац из оригинала данной главы, переведенный на итальянский язык, затем снова на английский с помощью службы перевода Systran.

Задача перевода является сложной, поскольку, вообще говоря, для ее решения требуется глубокое понимание текста, а для этого, в свою очередь, необходимо глубокое понимание ситуации, о которой идет речь. Это утверждение является справедливым применительно даже к очень простым текстам, в частности даже к “текстам”, состоящим из одного слова. Рассмотрим слово “Open” на двери магазина⁴. Оно со-

⁴ Этот пример предложил нам Мартин Кэй (Martin Kay).

общает, что в данный момент магазин принимает покупателей. Теперь предположим, что такое же слово “Open” написано на большом плакате рядом с вновь открытым магазином. Оно означает, что магазин уже работает, но люди, прочитавшие это сообщение, не почувствуют себя обманутыми, узнав, что магазин закрыли на ночь, а плакат оставили висеть на стенде. Дело в том, что в этих двух сообщениях одно и то же слово использовалось для передачи разных смысловых значений. С другой стороны, в тех магазинах, где с покупателями общаются на немецком языке, на дверях принято вешать табличку “Offen”, а на соответствующих плакатах, сообщающих об открытии нового магазина, писать слова “Neu Eröffnet”.

Таблица 23.4. Результаты двух последовательных применений средств грубого машинного перевода текста первого абзаца из оригинала настоящей главы

| Текст на итальянском языке | Текст на английском языке |
|--|---|
| In capitolo 22 abbiamo visto come un agente potrebbe comunicare con un altro agente (essere umano o software) che usando le espressioni in un linguaggio reciprocamente accordato. Completare sintattico e l'analisi semantica delle espressioni è necessaria da estrarre il significato completo del utterances ed è possibile perchè le espressioni sono corte e limitate ad un settore limitato | In chapter 22 we have seen as an agent could communicate with an other agent (to be human or software) that using the expressions in a language mutual come to an agreement. Complete syntactic and the semantic analysis of the expressions is necessary to extract the complete meant one of the utterances and is possible because the expressions short and are limited to a dominion |

Проблема состоит в том, что в различных языках мир подразделяется на категории по-разному. Например, французское слово “doux” охватывает широкий диапазон смысловых значений, приблизительно соответствующих английским словам “soft”, “sweet” и “gentle”. Аналогичным образом, английское слово “hard” охватывает практически все области применения немецкого слова “hart” (физически стойкий, жесткий), а также области использования слова “schwierig” (трудный). Немецкий глагол “heilen” охватывает все области применения английского существительного “cure” в медицине, а также области применения английского глагола “heal” в качестве транзитивного и нетранзитивного в обычном языке. Поэтому задача представления смысла предложения для перевода сложнее по сравнению с той ситуацией, когда эта задача решается в целях понимания смысла предложения на одном языке. В системе синтаксического анализа текста на одном языке могут использоваться предикаты наподобие $Open(x)$, а в случае перевода язык представления должен обеспечивать проведение более тонких различий, например, с учетом того, что предикат $Open_1(x)$ должен представлять смысл надписи “Offen”, а $Open_2(x)$ — смысл надписи “Neu Eröffnet”. Язык представления, в котором учитываются все различия, необходимые для представления целого ряда языков, называется **промежуточным языком**.

Чтобы выполнить беглый перевод, переводчик (человек или машина) должен прочитать первоначальный текст, понять тему, к которой он относится, и составить соответствующий текст на целевом языке, достаточно качественно описывающий ту же или аналогичную тему. При этом часто приходится брать на себя определенную ответственность. Например, английское слово “you”, обращенное к отдельному лицу, может быть переведено на французский язык либо как формальное обращение “vous”, либо как неформальное “tu”. Дело в том, что просто не существует способа, позволяющего перевести обращение “you” на французский язык, не приняв вместе с

тем решения о том, должно ли это обращение быть формальным или неформальным. Переводчики (и машины, и люди) иногда испытывают затруднения, принимая подобные решения.

Системы машинного перевода


Системы машинного перевода различаются по тому признаку, на каком уровне в них осуществляется анализ текста. В некоторых системах предпринимается попытка полностью проанализировать входной текст вплоть до получения представления на промежуточном языке (как было сделано в главе 22), а затем сформировать из этого представления предложения на целевом языке. Такая задача является сложной, поскольку она включает в качестве подзадачи проблему полного понимания языка, а к этому добавляются все сложности, связанные с применением промежуточного языка. К тому же такой подход является ненадежным, поскольку в случае неудачного завершения анализа не формируются также выходные данные. А его преимуществом является то, что нет таких частей системы, функционирование которых основано на знании одновременно двух языков. Это означает, что может быть создана система с промежуточным языком, позволяющая переводить тексты с одного из n языков на другой за счет трудозатрат, пропорциональных $O(n)$, а не $O(n^2)$.

Другие системы основаны на так называемом методе **передачи**. В них имеется база данных, состоящая из правил перевода (или примеров), а перевод осуществляется непосредственно путем согласования текста с правилами (или примерами). Передача может осуществляться на лексическом, синтаксическом или семантическом уровне. Например, строго синтаксическое правило преобразует английское словосочетание [*Adjective Noun*] (*adjective* — имя прилагательное, *noun* — имя существительное) во французское словосочетание [*Noun Adjective*]. А, допустим, смешанное синтаксическое и лексическое правило устанавливает соответствие между французским выражением [S_1 "et puis" S_2] и английским [S_1 "and then" S_2]. Передача, выполняемая с непосредственным преобразованием одного предложения в другое, известна под названием метода **перевода с помощью памяти**, поскольку она основана на запоминании большого множества пар (английский, французский). Метод передачи является надежным, поскольку он всегда обеспечивает выработку пусть даже каких-то выходных данных и при этом по меньшей мере хотя бы часть полученных слов обязательно оказывается правильной. На рис. 23.2 схематически показаны различные уровни передачи.

Статистический машинный перевод

В начале 1960-х годов широко распространилось мнение, что компьютеры вскоре смогут без особых проблем переводить с одного естественного языка на другой, в соответствии с тем, что в проекте Тьюринга удалось добиться успешного "перевода" закодированных сообщений на немецком языке в немецкий текст, доступный для восприятия. Но к 1966 году стало ясно, что для беглого перевода требуется понимание смысла сообщений, а для взлома кода — нет.

В последнее десятилетие наметилась тенденция к использованию систем машинного перевода, основанных на статистическом анализе. Безусловно, можно добиться выигрыша благодаря применению статистических данных и четкой вероятностной

модели того, в чем состоит качественный анализ или передача текста, на любом из этапов, показанных на рис. 23.2. Но под понятием “статистического машинного перевода” подразумевается общий подход к решению проблемы перевода, который основан на поиске наиболее вероятного перевода предложения с использованием данных, полученных из двуязычной совокупности текстов. В качестве примера двуязычной совокупности текстов можно назвать  **парламентские отчеты**⁵, которые представляют собой протоколы дебатов в парламенте. Двуязычные парламентские отчеты издаются в Канаде, Гонконге и других странах; официальные документы Европейского экономического сообщества издаются на 11 языках; а Организация объединенных наций публикует документы на нескольких языках. Как оказалось, эти материалы представляют собой бесценные ресурсы для статистического машинного перевода.

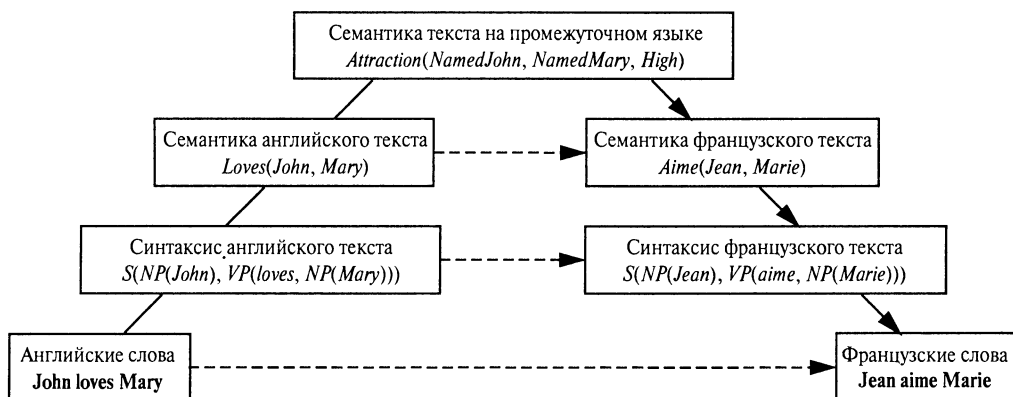


Рис. 23.2. Схематическое изображение вариантов организации систем машинного перевода. Схема начинается с английского текста, показанного в левой нижней части. Система с промежуточным языком следует по сплошным линиям, выполняя синтаксический анализ английского текста и преобразуя его вначале в синтаксическую форму, затем в семантическую форму представления и в форму представления на промежуточном языке, после этого выполняет этапы преобразования в семантическую, синтаксическую и лексическую форму на французском языке. В системе на основе передачи в качестве сокращенных путей используются пунктирные линии. В различных системах передача осуществляется на разных уровнях, причем в некоторых системах она происходит одновременно на нескольких уровнях

Проблему перевода английского предложения E , скажем, во французское⁶ предложение F можно представить в виде следующего уравнения, предусматривающего применение правила Байеса:

$$\begin{aligned}
 \underset{F}{\operatorname{argmax}} P(F|E) &= \underset{F}{\operatorname{argmax}} P(E|F) P(F) / P(E) \\
 &= \underset{F}{\operatorname{argmax}} P(E|F) P(F)
 \end{aligned}$$

⁵ В английском языке такие отчеты обозначаются словом Hansard в честь Уильяма Хансарда (William Hansard), который впервые опубликовал британские парламентские отчеты в 1811 году.

⁶ В данном разделе речь идет о задаче перевода с английского языка на французский. Старайтесь избегать путаницы, связанной с тем фактом, что правило Байеса требует от нас, чтобы мы рассматривали вероятность $P(E|F)$, а не $P(F|E)$, в результате чего создается впечатление, как будто перевод осуществляется с французского на английский.

Это правило указывает, что мы должны рассмотреть все возможные французские предложения F и выбрать из них то, которое максимизирует произведение $P(E|F)P(F)$. Коэффициент $P(E)$ можно проигнорировать, поскольку он является одинаковым для любого F . Коэффициент $P(F)$ представляет собой **языковую модель** для французского языка; он указывает, насколько велика вероятность появления данного конкретного предложения во французском тексте. Вероятность $P(E|F)$ представляет собой **модель перевода**; она указывает, насколько велика вероятность того, что некоторое английское предложение будет использоваться в качестве перевода, если дано определенное французское предложение.

Внимательного читателя, безусловно, заинтересует вопрос о том, чего мы добьемся, определив вероятность $P(F|E)$ в терминах $P(E|F)$. В других областях применения правила Байеса такая перестановка термов в выражениях для условной вероятности была сделана в связи с тем, что мы стремились перейти к использованию причинной модели. Например, для вычисления вероятности наличия определенных симптомов при определенном заболевании, $P(Disease|Symptoms)$, применялась причинная модель $P(Symptoms|Disease)$. В отличие от этого при переводе с одного языка на другой ни одно из направлений перевода не характеризуется большей причинной зависимостью, чем другое. В данном случае правило Байеса применяется в связи с тем, что мы, по-видимому, сможем легко определить с помощью обучения языковую модель $P(F)$, которая является более точной по сравнению с моделью перевода $P(E|F)$ (а также более точной по сравнению с непосредственно полученной оценкой $P(F|E)$). По сути такой подход позволяет разделить задачу на две части — вначале применить модель перевода $P(F|E)$ для поиска подходящих французских предложений, в которых упоминаются те же понятия, что и в английском предложении (но это не обязательно должны быть французские предложения, полностью адекватные английскому предложению); затем воспользоваться языковой моделью $P(F)$ (для которой имеются намного лучшие оценки вероятностей), чтобы выбрать наиболее подходящий вариант перевода.

В качестве **языковой модели** $P(F)$ может использоваться любая модель, позволяющая присвоить предложению определенное значение вероятности. При наличии очень большой совокупности текстов можно оценить $P(F)$ непосредственно путем подсчета количества случаев появления каждого предложения в этой совокупности текстов. Например, если с помощью Web будет собрано 100 миллионов французских предложений и обнаружено, что предложение “Clique ici” (Щелкните здесь) появляется 50 тысяч раз, то $P(\text{"Clique ici"})$ равно 0,0005. Но даже при наличии 100 миллионов примеров количество экземпляров большинства возможных предложений будет равно нулю⁷. Поэтому мы будем использовать знакомую языковую модель двухсловных сочетаний, в которой вероятность французского предложения, состоящего из слов $f_1 \dots f_n$, может быть представлена следующим образом:

⁷ Даже если в словаре имеется только 100 тысяч слов, то 99,99999% возможных предложений, состоящих из трех слов, будут присутствовать в совокупности текстов из 100 миллионов предложений в количестве, равном нулю. По мере увеличения длины предложений ситуация становится еще хуже.

$$P(f_1 \dots f_n) = \prod_{i=1}^n P(f_i | f_{i-1})$$

Для этого необходимо знать вероятности двухсловных сочетаний, такие как $P(\text{"Eiffel"} | \text{"tour"}) = .02$. Эти данные позволяют учитывать только самые локальные проявления синтаксических связей, в которых слово зависит лишь от предыдущего слова. Но этого часто достаточно для грубого перевода⁸.

Задача выбора **модели перевода**, $P(E|F)$, является более сложной. С одной стороны, отсутствует готовая коллекция пар предложений (английский, французский), с помощью которой можно было бы проводить обучение. С другой стороны, такая модель сложнее, поскольку в ней рассматривается перекрестное произведение предложений, а не просто отдельные предложения. Начнем с одной чрезмерно упрощенной модели перевода и постепенно усовершенствуем ее до такого уровня, чтобы она напоминала известную разработку IBM Model 3 [196], которая все еще может показаться чрезмерно упрощенной, но обнаружила свою способность вырабатывать приемлемые варианты перевода примерно в половине случаев.

Рассматриваемая чрезмерно упрощенная модель перевода основана на таком принципе: “Чтобы перевести предложение, просто переведите каждое слово отдельно, независимо от другого, в порядке слева направо”. Это — модель выбора одно-слового сочетания. Она позволяет легко вычислить вероятность перевода:

$$P(E|F) = \prod_{i=1}^n P(E_i | F_i)$$

В некоторых случаях эта модель действует безукоризненно. Например, рассмотрим следующую конструкцию:

$$P(\text{"the dog"} | \text{"le chien"}) = P(\text{"the"} | \text{"le"}) \times P(\text{"dog"} | \text{"chien"})$$

При любом обоснованном подборе вариантов значений вероятностей выражение “the dog” (собака) будет служить наиболее правдоподобным переводом выражения “le chien”. Но в большинстве случаев прямолинейные попытки применения этой модели оканчиваются неудачей. Одна из проблем связана с порядком слов. Английское слово “dog” соответствует французскому слову “chien”, а понятие, обозначаемое в английском языке словом “brown” (коричневый), во французском языке обозначается словом “brun”. Однако словосочетание “brown dog” переводится как “chien brun”. Еще одна проблема состоит в том, что словесные обороты не связаны друг с другом в форме взаимно однозначного соответствия. Английское слово “home” часто переводят с помощью выражения “à la maison”, поэтому имеет место соответствие “один к трем” (или три к одному, при противоположном направлении перевода). Невзирая на наличие указанных проблем, разработчики модели IBM Model 3 приняли за основу жесткий

⁸ Если в переводе нужно передать более тонкие нюансы, то модель, основанная на $P(f_i | f_{i-1})$, безусловно, становится неприемлемой. В качестве одного из известных примеров можно указать, что знаменитый цикл романов “A la recherche du temps perdu” (В поисках утраченного времени) Марселя Пруста объемом в 3500 страниц начинается и оканчивается одним и тем же словом, поэтому некоторые переводчики решили сделать то же самое и построили весь свой перевод на одном слове, находящемся примерно за 2 миллиона слов от него.

подход, по сути базирующийся на модели однословных сочетаний, но ввели несколько дополнений для компенсации ее недостатков.

Для того чтобы можно было учесть тот факт, что некоторые слова не допускают перевода один к одному, в эту модель было введено понятие Δ **фертильности** (fertility — плодovitость) слова. Слово с фертильностью n копируется n раз, после чего каждая из этих n копий переводится независимо. Модель содержит параметры, которые показывают значение $P(\text{Fertility}=n|\text{word})$ для каждого французского слова. Для перевода выражения “à la maison” как выражения “home” в этой модели необходимо выбрать фертильность 0 для “à” и “la” и фертильность 1 для “maison”, а затем применить модель перевода однословных сочетаний, чтобы перевести “maison” как “home”. Такой подход кажется достаточно приемлемым, поскольку “à” и “la”, будучи словами с низким информационным содержанием, могут быть на полном основании заменены в процессе перевода пустой строкой. Но применение такого метода для перевода в другом направлении становится более сомнительным. Слову “home” должна быть назначена фертильность 3, что приведет к его преобразованию в “home home home”. Тогда первое слово “home” должно быть переведено как “à”, второе как “la” и третье как “maison”. Но с точки зрения этой модели перевода выражение “à la maison” должно иметь точно такую же вероятность, как “maison la à” (в этом и состоит та часть данного подхода, которая может быть поставлена под сомнение.) Дело в том, что выбор того или иного варианта должен осуществляться на уровне языковой модели. Может показаться, что было бы более целесообразным применение непосредственного перевода слова “home” как выражения “à la maison” вместо использования косвенного варианта с преобразованием в “home home home”, но для этого потребовалось бы больше параметров и их было бы труднее получить из доступной совокупности текстов.

В последней части этой модели перевода предусмотрена перестановка слов в правильные позиции. Такая перестановка осуществляется с помощью модели смещений, в которой указано, как следует перемещать слова из их первоначальных позиций в окончательные позиции. Например, при переводе “chien brun” как “brown dog” слово “brown” получает смещение +1 (оно сдвигается на одну позицию вправо), а слово “dog” получает смещение -1. На первый взгляд может показаться, что смещение должно быть зависимым от слова, например, такие прилагательные, как “brown”, как правило, должны иметь положительное смещение, поскольку во французском языке прилагательные обычно стоят после существительных. Но разработчики модели IBM Model 3 решили, что для реализации подхода, в котором смещения зависят от слова, потребуется слишком много параметров, поэтому смещение должно быть независимым от слова и зависимым только от положения внутри предложения, а также от длины предложений на обоих языках. Это означает, что в этой модели осуществляется оценка следующих параметров:

$$P(\text{Offset}=o|\text{Position}=p, \text{EngLen}=m, \text{FrLen}=n)$$

Таким образом, для определения смещения слова “brown” в выражении “brown dog” с помощью базы данных определяется значение $P(\text{Offset}|1, 2, 2)$, что может, например, привести к получению значения +1 с вероятностью 0,3 и 0 с вероятностью 0,7. Но такая модель смещений кажется еще более сомнительной, особенно тем, кто, например, пытался составить надпись из букв с магнитами на своем холодильнике и понял, что это намного сложнее, чем высказать то же самое с помощью

обычной речи. Вскоре будет показано, что такое решение было принято разработчиками не потому, что оно основано на качественной модели языка, а в связи с тем, что обеспечивает эффективное использование имеющихся данных. Так или иначе модель смещения наглядно показывает, что модель перевода среднего качества может быть значительно улучшена с помощью высококачественной языковой модели для французского языка. Ниже приведен пример, показывающий все этапы перевода одного предложения.

| | | | | | | | | | | |
|---------------------------------------|-----|-------|-------|-----|-----|-----|------|---|--------|--------|
| Исходный французский текст: | Le | chien | brun | n' | est | pas | allé | à | la | maison |
| Модель фертильности: | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 1 |
| Трансформированный французский текст: | Le | chien | brun | n' | est | | allé | | maison | |
| Модель выбора слов: | The | dog | brown | not | did | | go | | home | |
| Модель смещений: | 0 | +1 | -1 | +1 | -1 | | 0 | | 0 | |
| Целевой английский текст: | The | brown | dog | did | not | | go | | home | |

Теперь нам известно, как рассчитать вероятность $P(F|E)$ для любой пары предложений (французский, английский). Но в действительности перед нами стоит задача, получив некоторое английское предложение, найти французское предложение, которое максимизирует эту вероятность. Для этого недостаточно просто перебирать предложения, поскольку если предположить, что количество слов во французском языке равно 10^5 , то существует 10^{5n} предложений длины n , а также много вариантов каждого из этих предложений. И даже если будут рассматриваться только 10 наиболее часто встречающихся вариантов дословного перевода для каждого слова и учитываться лишь смещения 0 или ± 1 , все равно будет получено около $2^{n/2} 10^n$ предложений, а это означает, что может быть выполнен их полный перевод при $n=5$, но не при $n=10$. Поэтому вместо перебора необходимо осуществлять поиск наилучшего решения. Практика показала, что эффективным является поиск на основе алгоритма A^* ; см. [545].

Определение с помощью обучения вероятностей для машинного перевода

Выше была кратко описана модель для $P(F|E)$, которая предусматривает применение четырех перечисленных ниже множеств параметров.

- Языковая модель. $P(word_i | word_{i-1})$.
- Модель фертильности. $P(Fertility=n | word_F)$.
- Модель выбора слова. $P(word_E | word_F)$.
- Модель смещения. $P(Offset=o | pos, len_E, len_F)$.

Но даже при использовании скромного словаря, состоящего из 1000 слов, для этой модели требуются миллионы параметров. Очевидно, что необходимо обеспечить определение этих параметров с помощью обучения на основе данных. Предположим, что единственными доступными данными является двуязычная совокупность текстов. Ниже описан способ использования этих данных.

- Сегментация на предложения. Единицей перевода является предложение, поэтому нам потребуется разбить совокупность текстов на предложения. Надежным показателем конца предложения является точка, но в таком фрагменте

текста, как “Dr. J. R. Smith of Rodeo Dr. arrived.”, признаком конца предложения является только последняя точка. Сегментация на предложения может быть выполнена с точностью около 98%.

- Оценка языковой модели для французского языка $P(word_i | word_{i-1})$. Рассматривая только французскую половину совокупности текстов, подсчитать частоты пар слов и выполнить выравнивание, чтобы получить оценку $P(word_i | word_{i-1})$. Например, может быть получено значение $P("Eiffel" | "tour") = .02$.
- Выравнивание предложений. Для каждого предложения в английской версии определить, какое предложение (предложения) соответствует ему во французской версии. Обычно следующее предложение в английском тексте соответствует следующему предложению во французском тексте в форме согласования “один к одному”, но иногда возникают другие варианты: одно предложение на одном из языков может быть разбито на два, что приводит к согласованию “два к одному”, или может быть изменен на противоположный порядок следования двух предложений, а это приведет к согласованию “два к двум”. Выравнивание предложений (“один к одному”, “один к двум” или “два к двум” и т.д.) может быть обеспечено только на основании сравнения длины предложений с точностью в пределах от 90 до 99% с использованием одного из вариантов алгоритма сегментации Витерби (см. листинг 23.1). С применением отметок, общих для обоих языков, таких как числа или имена собственные, а также слов, которые, как известно, имеют в двуязычном словаре однозначный перевод, можно добиться еще лучшего выравнивания.

Теперь можно приступить к оценке параметров модели перевода. Такую задачу можно решить, приняв довольно слабое начальное предположение, а затем постепенно его улучшая, как описано ниже.

- Оценка начальной модели фертильности $P(Fertility=n | word_F)$. Найдя французское предложение длины m , которое выравнивается с английским предложением длины n , будем рассматривать его как свидетельство того, что каждое французское слово имеет фертильность n/m . Рассмотрим все свидетельства во всех предложениях, чтобы получить распределение вероятностей фертильности для каждого слова.
- Оценка начальной модели выбора слова $P(word_E | word_F)$. Рассмотрим все французские предложения, которые содержат, скажем, слова “brun”. Слова, которые появляются наиболее часто в английских предложениях, выравниваемых с этими предложениями, являются наиболее вероятными буквальными переводами слова “brun”.
- Оценка начальной модели смещения $P(Offset=o | pos, len_E, len_F)$. Теперь, после получения модели выбора слова, воспользуемся ею, чтобы получить оценку модели смещения. Для каждого английского предложения длины n , которая выравнивается с французским предложением длины m , проанализировать каждое французское слово в предложении (в позиции i) и каждое английское слово в предложении (в позиции j), которое является наиболее вероятным вариантом выбора для французского слова, и рассматривать его как свидетельство для вероятности $P(Offset=i-j | i, n, m)$.

- Усовершенствование всех оценок. Воспользоваться алгоритмом ЕМ (expectation-maximization — ожидание-максимизация), чтобы усовершенствовать оценки. Скрытой переменной является **вектор выравнивания слов** между парами предложений, выровненными по предложениям. Этот вектор указывает для каждого английского слова позицию соответствующего французского слова во французском предложении. Например, может быть получено следующее:

| | | | | | | | | | | |
|-----------------------------|-----|-------|------|-----|-----|-----|------|---|----|--------|
| Исходный французский текст: | Le | chien | brun | n' | est | pas | allé | à | la | maison |
| Целевой английский текст: | The | brown | dog | did | not | go | home | | | |
| Вектор выравнивания слов: | 1 | 3 | 2 | 5 | 4 | 7 | 10 | | | |

Вначале с использованием текущих оценок параметров создадим вектор выравнивания слов для каждой пары предложений. Это позволит нам получать лучшие оценки. Модель фертильности оценивается путем подсчета того, сколько раз один из элементов вектора выравнивания слов указывает на несколько слов или на нулевое количество слов. После этого в модели выбора слов могут рассматриваться только те слова, которые выровнены друг с другом, а не все слова в предложении, тогда как в модели смещений может рассматриваться каждая позиция в предложении для определения того, насколько часто она смещается в соответствии с вектором выравнивания слов. К сожалению, точно не известно, каковым является правильное выравнивание, а количество вариантов выравнивания слишком велико для того, чтобы перебрать их все. Поэтому мы вынуждены осуществлять поиск выравниваний с высокой вероятностью и взвешивать их по их вероятностям, собирая свидетельства для новых оценок параметров. Это все, что требуется для алгоритма ЕМ. На основании начальных параметров вычисляются выравнивания, а с помощью выравниваний уточняются оценки параметров. Такая процедура повторяется до полной сходимости.

23.5. РЕЗЮМЕ

Основные положения, изложенные в этой главы, перечислены ниже.

- Вероятностные языковые модели, основанные на n -элементных сочетаниях, позволяют получить весьма значительный объем информации о языке.
- Контекстно-свободные грамматики (Context-Free Grammar — CFG) могут быть расширены до вероятностных контекстно-свободных грамматик, которые позволяют проще определять их параметры с помощью обучения из имеющихся данных, а также легче решать задачу устранения неоднозначности.
- В системах **информационного поиска** используется очень простая языковая модель, основанная на обработке мультимножеств слов, но даже эта модель позволяет достичь высоких показателей **полноты** и **точности** на очень больших совокупностях текстов.
- В системах **извлечения информации** используется более сложная модель, которая включает простейшие синтаксические и семантические конструкции. Для реализации таких систем часто применяются каскады конечных автоматов.
- В практически применяемых системах **машинного перевода** используется целый ряд методов, начиная от полного синтаксического и семантического анализа и заканчивая статистическими методами, основанными на учете частот слов.

- При формировании статистической языковой системы лучше всего опереться на модель, позволяющую эффективно использовать имеющиеся данные, даже если эта модель кажется чрезмерно упрощенной.

БИБЛИОГРАФИЧЕСКИЕ И ИСТОРИЧЕСКИЕ ЗАМЕТКИ

Подход с применением моделей *n*-буквенных сочетаний для моделирования языка был предложен Марковым [983]. Клод Шеннон [1394] впервые создал модели *n*-словных сочетаний английского языка. Хомский [250], [251] указал на ограничения моделей на основе конечных автоматов по сравнению с моделями на основе контекстно-свободных грамматик и пришел к заключению: “Вероятностные модели не позволяют каким-то образом добиться лучшего понимания некоторых основных проблем синтаксической структуры”. Это утверждение справедливо, но в нем игнорируется тот факт, что вероятностные модели обеспечивают лучшее понимание некоторых других основных проблем, а именно тех проблем, которые не могут быть решены с помощью контекстно-свободных грамматик. Замечания, сделанные Хомским, оказали неблагоприятный эффект, выразившийся в том, что в течение двух десятилетий многие исследователи избегали использования статистических моделей. Положение изменилось лишь после того, как указанные модели снова вышли на передний план и стали применяться при распознавании речи [730].

Метод сглаживания с добавлением единицы был предложен Джеффри [728], а метод сглаживания с удалением путем интерполяции разработан Елинеком и Мерсером [732], которые использовали этот метод для распознавания речи. В число других методов входят сглаживание Виттена–Белла [1605] и сглаживание Гуд–Тьюринга [257]. Последний метод также широко применяется при решении задач биоинформатики. Проблематика биостатистических и вероятностных задач NLP постепенно сближается, поскольку в каждой из этих областей приходится иметь дело с длинными структурированными последовательностями, выбранными из алфавита непосредственных составляющих.

Простые модели *n*-буквенных и *n*-словных сочетаний не являются единственными возможными вероятностными моделями. В [136] описана вероятностная модель текста, называемая **скрытым распределением Дирихле**, в которой документ рассматривается как комбинация тем, а каждая из тем характеризуется собственным распределением слов. Эта модель может рассматриваться как дополнение и уточнение модели **скрытой семантической индексации** Дирвестера [376] (см. также [1169]); кроме того, она тесно связана с моделью сочетания многочисленных причин [1345].

В **вероятностных контекстно-свободных грамматиках** (Probabilistic Context-Free Grammar — PCFG) устранены все недостатки вероятностных моделей, отмеченные Хомским, и они показали свои преимущества над обычными контекстно-свободными грамматиками. Грамматики PCFG были исследованы Бутом [151] и Саломеа [1346]. В [729] представлен алгоритм декодирования стека, представляющий собой один из вариантов алгоритма поиска Витерби, который может использоваться для определения наиболее вероятной версии синтаксического анализа с помощью грамматики PCFG. В [63] представлен внешний–внутренний алгоритм, а в [889] описаны области его применения и ограничения. В [236] и [804] обсуждаются проблемы синтаксического анализа с помощью грамматик в виде **банка деревьев**.

В [1467] показано, как определять с помощью обучения грамматические правила на основе слияния байесовских моделей. Другие алгоритмы для грамматик PCFG представлены в [235] и [980]. В [282] приведен обзор результатов, полученных в этой области, и даны пояснения к одной из наиболее успешных программ статистического синтаксического анализа.

К сожалению, грамматики PCFG при выполнении самых различных задач показывают более низкую производительность по сравнению с простыми n -элементными моделями, поскольку грамматики PCFG не позволяют представить информацию, связанную с отдельными словами. Для устранения этого недостатка некоторые авторы [281], [237], [713] предложили варианты **лексикализованных вероятностных грамматик**, в которых совместно используются контекстно-свободные грамматики и статистические данные, касающиеся отдельных слов.

Первой попыткой собрать сбалансированную совокупность текстов для эмпирической лингвистики явилось создание коллекции Brown Corpus [493]. Эта совокупность состояла примерно из миллиона слов с отметками, обозначающими части речи. Первоначально эта коллекция хранилась на 100 тысячах перфокарт. Банк деревьев синтаксического анализа Пенна [982] представляет собой коллекцию, состоящую примерно из 1,6 миллиона слов текста, для которого вручную выполнен синтаксический анализ с преобразованием в деревья. Эта коллекция помещается на компакт-диске. В издании British National Corpus [905] данная коллекция была расширена до 100 миллионов слов. В World Wide Web хранится свыше триллиона слов больше чем на 10 миллионах серверов.

В последнее время растет интерес к области **информационного поиска**, обусловленный широким применением поиска в Internet. В [1296] приведен обзор ранних работ в этой области и представлен принцип ранжирования вероятностей. В [980] дано краткое введение в проблематику информационного поиска в контексте статистических подходов к решению задач NLP. В [59] приведен обзор общего назначения, заменивший более старые классические работы [492] и [1347]. Книга *Managing Gigabytes* [1606] посвящена решению именно той задачи, о которой говорит ее название, — описанию того, как можно эффективно индексировать, применять сжатие и выполнять запросы применительно к совокупности текстов гигабайтовых размеров. В рамках конференции TREC, организованной Национальным институтом стандартов и технологии (National Institute of Standards and Technology — NIST) при правительстве Соединенных Штатов, проводятся ежегодные соревнования между системами информационного поиска и публикуются труды с описанием достигнутых результатов. За первые семь лет таких соревнований производительность участвующих в них программ выросла примерно в два раза.

Наиболее широко применяемой моделью для информационного поиска является **модель векторного пространства** Салтона [1348]. В первые годы развития этой области указанная работа Салтона была фактически самой влиятельной. Имеются также две альтернативные вероятностные модели. Модель, представленная в этой книге, основана на [1225]. В ней моделируется совместное распределение вероятностей $P(D, Q)$ в терминах $P(Q|D)$. В другой модели [985], [1297] используется вероятность $P(D|Q)$. В [879] показано, что обе эти модели основаны на одном и том же совместном распределении вероятностей, но от выбора модели зависит то, какие методы должны применяться для определения параметров с помощью обучения. Описание,

приведенное в данной главе, основано на обеих этих моделях. В [1522] приведено сравнение различных моделей информационного поиска.

В [187] описана реализация машины поиска для World Wide Web, включая алгоритм PageRank, в основе которого лежит независимый от запроса критерий качества документа, базирующийся на анализе Web-ссылок. В [805] описано, как находить авторитетные источники информации в Web с использованием анализа ссылок. В [1411] приведены результаты исследования журнала с данными о миллиарде поисковых операций, выполненных в Web. В [864] приведен обзор литературы по исправлению орфографических ошибок. В [1230] описан классический алгоритм выделения основы с помощью правил, а в [860] описан вариант, в котором применяется словарь.

В [980] приведен хороший обзор проблематики классификации и кластеризации документов. В [738] используются теория статистического обучения и теория машин векторов поддержки для теоретического анализа ситуаций, в которых классификация должна быть успешной. В [37] приведены данные о том, что при классификации новостных сообщений агентства Reuters, относящихся к категории “Earnings” (Доходы), была достигнута точность 96%. В [824] приведены данные о том, что при использовании наивного байесовского классификатора достигается точность вплоть до 95%, а при использовании байесовского классификатора, в котором учитываются некоторые зависимости между характеристиками, — вплоть до 98,6%. В [922] приведен обзор результатов, достигнутых за сорок лет применения наивных байесовских моделей для классификации и поиска в тексте.

Последние достижения в этой области публикуются в журнале *Information Retrieval* и в трудах ежегодной конференции *SIGIR*.

Одними из первых программ извлечения информации являются Gus [143] и Frump [380]. В основе некоторых проектов современных систем извлечения информации лежат работы в области семантических грамматик, проводившиеся в 1970-х и 1980-х годах. Например, в интерфейсе системы резервирования авиабилетов с семантической грамматикой используются такие категории, как *Location* (место нахождения) и *FlyTo* (место назначения), а не *NP* и *VP*. Описание результатов реализации одной из систем, основанных на семантических грамматиках, приведено в [130].

Новейшие результаты исследований по извлечению информации пропагандируются на ежегодных конференциях MUC (Message Understanding Conference), спонсором которых выступает правительство США. Система FASTUS была разработана Хоббсом и др. [664]; в сборнике статей, в котором впервые была опубликована информация об этой системе [1299], можно найти информацию и о других системах, в которых используются модели конечных автоматов.


В 1930-м году Петр Троянский (Petr Troyanskii) подал заявку на патент, в котором была сформулирована идея “машины перевода”, но в то время еще не существовали компьютеры, позволяющие реализовать эту идею. В марте 1947 года Уоррен Вивер (Warren Weaver), сотрудник Фонда Рокфеллера, написал Норберту Винеру письмо, в котором указал, что решение задачи машинного перевода вполне возможно. Опираясь на работы в области криптографии и теории информации, Вивер писал: “Когда я рассматриваю статью, написанную на русском языке, я говорю себе: «Она фактически написана на английском языке, но закодирована странными символами. Теперь я приступаю к ее декодированию»”. В течение следующего десятилетия все сообщество специалистов в этой области предпринимало упорные попытки декодирования текстов на иностранном языке таким способом. Компания IBM про-

демонстрировала соответствующую зачаточную систему в 1954 году. Энтузиазм, характерный для этого периода, показан в [69] и [942]. Последующее разочарование в возможностях машинного перевода описано Линдсеем [935], указавшим также на некоторые препятствия, связанные с необходимостью обеспечения взаимодействия синтаксиса и семантики, а также с потребностью в наличии знаний о мире, с которыми сталкивается машинный перевод. Правительство США выразило недовольство полным отсутствием прогресса в этой области и сформулировало свое заключение в одном из отчетов, который известен как отчет ALPAC [21]: “Нет ни ближайших, ни обозримых перспектив создания практически применимых систем машинного перевода”. Однако работы в ограниченном объеме продолжались, и в ВВС США в 1970 году была развернута система Systran, которая была взята на вооружение Европейским экономическим сообществом в 1976 году. В том же 1976 году была развернута система перевода сообщений о погоде Taum-Meteo [1255]. К началу 1980-х годов возможности компьютеров возросли до такой степени, что выводы отчета ALPAC потеряли свою актуальность. В [1548] приведены сведения о некоторых новейших приложениях машинного перевода, основанных на системе Wordnet. Учебное введение в эту область приведено в [710].

Первые предложения по использованию статистического машинного перевода были сделаны в заметках Уоррена Вивера, опубликованных в 1947 году, но возможность практического применения этих методов появилась только в 1980-х годах. Описание этой тематики, приведенное в данной главе, основано на работе Брауна и его коллег из компании IBM [195], [196]. Эти труды весьма насыщены математической символикой, поэтому прилагаемый к ним учебник Кевина Найта [806] воспринимается как глоток свежего воздуха. В более современных исследованиях по статистическому машинному переводу наблюдается отказ от модели двухсловных сочетаний в пользу моделей, которые включают некоторые синтаксические конструкции [1627]. Первые работы в области сегментации предложений были выполнены Палмером и Херстом [1166]. Задача выравнивания двуязычных предложений рассматривается в [1042].


Есть две превосходные книги по вероятностной обработке лингвистической информации: книга [235] является краткой и точной, а книга [980] — всеобъемлющей и современной. С состоянием работ по созданию практических методов обработки лингвистической информации можно ознакомиться по материалам проводимой один раз в два года конференции *Applied Natural Language Processing* (ANLP) и конференции *Empirical Methods in Natural Language Processing* (EMNLP), а также по публикациям в журнале *Natural Language Engineering*. Организация SIGIR финансирует выпуск информационного бюллетеня и проведение ежегодной конференции по информационному поиску.

УПРАЖНЕНИЯ

- 23.1.  (Адаптировано из [756].) В этом упражнении предлагается разработать классификатор для выявления авторства: при наличии некоторого текста этот классификатор должен попытаться определить, какой из двух возможных авторов написал этот текст. Получите образцы текста двух различных авторов. Разделите их на обучающие и контрольные множества. После этого определи-

те с помощью обучения параметры модели однословных сочетаний для каждого автора по обучающему множеству. Наконец, для каждого контрольного множества рассчитайте его вероятность в соответствии с каждой моделью однословных сочетаний и присвойте эту вероятность наиболее вероятной модели. Оцените точность этого метода. Можете ли вы повысить его точность с помощью дополнительных характеристик? Эта подобласть лингвистики называется **стилиметрией**; к числу достижений в этой области относится идентификация автора “Заметок федералиста” (Federalist Papers) [1091] и некоторых произведений Шекспира, авторская принадлежность которых некогда оспаривалось [486].

- 23.2. В этом упражнении исследуется качество моделей n -элементных сочетаний, характерных для некоторого языка. Найдите или создайте моноязыковую совокупность, состоящую примерно из 100 тысяч слов. Сегментируйте ее на слова и вычислите частоту каждого слова. Каково количество присутствующих в ней различных слов? Начертите график зависимости частоты слов от их ранга (первое, второе, третье...) с логарифмической шкалой по горизонтали и по вертикали. Кроме того, подсчитайте частоты двухсловных сочетаний (два подряд идущих слова) и трехсловных сочетаний (три подряд идущих слова). Воспользуйтесь этими частотами для генерации языка: на основании моделей одно-, двух- и трехсловных сочетаний последовательно сформируйте образцы текста из 100 слов, выполняя выбор случайным образом в соответствии со значениями частот. Сравните три сформированных текста с фактически имеющимся текстом на рассматриваемом языке. Наконец, рассчитайте показатель связности каждой модели.
- 23.3. В этом упражнении рассматривается задача распознавания нежелательной электронной почты (спам). *Спамом* принято называть незатребованные объемистые коммерческие сообщения, поступающие по электронной почте. Утомительную задачу разборки спама приходится решать многим пользователям, поэтому создание надежного способа его устранения явилось бы большим достижением. Создайте две совокупности текстов — состоящую из почтовых сообщений, представляющих собой спам, и состоящую из обычных почтовых сообщений. Исследуйте каждую совокупность и определите, какие характеристики, скорее всего, окажутся применимыми для классификации: однословные сочетания, двухсловные сочетания, длина сообщений, отправитель, время получения и т.д. Затем проведите обучение алгоритма классификации (дерева решений, наивной байесовской модели или какого-то другого выбранного вами алгоритма) на обучающем множестве и определите его точность на контрольном множестве.
- 23.4. Создайте контрольное множество из пяти запросов и предъявите эти запросы трем основным машинам поиска Web. Оцените каждую из них по показателю точности для 1, 3 и 10 возвращенных документов и по среднему обратному рангу. Попытайтесь объяснить обнаруженные различия.
- 23.5. Попытайтесь определить, в какой из машин поиска, рассматриваемых в предыдущем упражнении, используются методы приведения к нижнему регистру, выделения основы, выявления синонимов и исправления орфографических ошибок.

- 23.6. Оцените, какой объем памяти является необходимым для индекса к совокупности Web-страниц, состоящей из миллиарда страниц. Укажите, какие предположения были вами приняты.
- 23.7.  Напишите регулярное выражение или короткую программу для извлечения названий компаний. Проверьте ее на совокупности, состоящей из деловых новостных сообщений. Определите полноту и точность полученных результатов.
- 23.8. Выберите пять предложений и передайте их в оперативную службу перевода. Переведите их с английского на другой язык, а затем снова на английский. Оцените, насколько полученные при этом предложения являются грамматически правильными и сохранившими смысл. Повторите этот процесс; будут ли во второй итерации получены худшие результаты или такие же результаты? Влияет ли на качество результатов выбор промежуточного языка?
- 23.9. Соберите некоторые примеры выражений с указанием времени, таких как “two o'clock”, “midnight” и “12:46”. Кроме того, подготовьте некоторые примеры, являющиеся грамматически неправильными, такие как “thirteen o'clock” или “half past two fifteen”. Напишите грамматику для языка выражений с указанием времени.
- 23.10. (*Адаптировано из [806].*) В модели машинного перевода IBM Model 3 предполагается, что после того как с помощью модели выбора слов будет подготовлен список слов, а с помощью модели смещения будут подготовлены возможные перестановки слов, можно будет применить языковую модель для выбора наилучшей перестановки. Данное упражнение посвящено исследованию того, насколько обоснованным является указанное предположение. Попытайтесь переставить слова в приведенных ниже предложениях, подготовленных с помощью модели IBM Model 3, в правильном порядке.
- have programming a seen never I language better
 - loves john mary
 - is the communication exchange of intentional information brought by about the production perception of and signs from drawn a of system signs conventional shared

С какими предложениями вам удалось справиться? Знания какого типа пришлось вам для этого привлечь? Проведите обучение модели двухсловных сочетаний с помощью обучающей совокупности и воспользуйтесь этой моделью для поиска перестановок некоторых предложений из контрольной совокупности с наибольшей вероятностью. Определите точность этой модели.

- 23.11. Согласно данным англо-французского словаря, переводом для слова “hear” является глагол “entendre”. Но если проводится обучение модели IBM Model 3 по отчетам канадского парламента, то наиболее вероятным переводом для слова “hear” становится “Bravo”. Объясните, почему такое происходит, и оцените, каким может быть распределение фертильности для слова “hear”. (*Подсказка.* Вам может потребоваться ознакомиться с каким-то текстом парламентского отчета. Попробуйте выполнить поиск в Web с помощью запроса [Hansard hear].)

24 ВОСПРИЯТИЕ

В данной главе речь идет о том, как ввести в компьютер исходные, необработанные данные, полученные из реального мира.

✎ **Восприятие** предоставляет агентам информацию о мире, в котором они обитают. Восприятие осуществляется с помощью ✎ **датчиков**. Датчиком может быть любое устройство, позволяющее зафиксировать состояние какого-то аспекта среды и передать полученные данные в качестве входных в программу агента. Датчик может быть настолько простым, как однобитовый детектор, который лишь определяет, разомкнут или замкнут выключатель, или настолько сложным, как сетчатка человеческого глаза, которая содержит больше ста миллионов фоточувствительных элементов. В данной главе наше внимание будет сосредоточено на зрении, поскольку оно намного превосходит по информативности все остальные чувства, когда приходится сталкиваться с проявлениями физического мира.

24.1. ВВЕДЕНИЕ

В распоряжении искусственных агентов имеется целый ряд *сенсорных модальностей*. К числу тех из них, которые являются общими и для людей, и для роботов, относятся зрение, слух и осязание. Проблема слухового восприятия, по крайней мере, в той части, которая касается восприятия речи, рассматривалась в разделе 15.6. Осязание, или **тактильное восприятие**, будет рассматриваться в главе 25, где речь идет о его использовании в сложнейших манипуляциях, выполняемых роботами, а остальная часть настоящей главы будет посвящена зрению. Некоторые роботы способны воспринимать модальности, не доступные людям, не пользующимся специальными приспособлениями, такие как радиоволны, инфракрасные лучи, сигналы глобальной системы навигации и определения положения (Global Positioning System — GPS) и другие беспроводные сигналы. Некоторые роботы осуществляют **активное восприятие**; это означает, что они посылают такие импульсные сигналы, как радарные или ультразвуковые, и принимают отражение этих импульсов от среды.

Существуют два способа, с помощью которых агент может использовать полученные им результаты восприятия. В подходе, основанном на **извлечении характеристик**, агенты распознают какое-то небольшое количество характеристик в полученных ими сенсорных входных данных и передают эти данные непосредственно в свою

программу агента, которая может вырабатывать команды по осуществлению действий, представляющих собой реакцию на изменение этих характеристик, или применять эти данные в сочетании с другой информацией. По такому принципу действовал агент в мире вампуса, оборудованный пятью датчиками, каждый из которых извлекал информацию об одной однобитовой характеристике. Кроме того, недавно стало известно, что в нервной системе мухи извлекаются данные о характеристиках из оптического потока и эти данные направляются прямо на мускулы, которые помогают мухе управлять своим движением в воздухе, что дает ей возможность быстро реагировать и изменять направление полета в течение 30 миллисекунд.

Альтернативным этому подходу является подход **на основе модели**, в котором сенсорные стимулы используются для реконструкции модели мира. При этом подходе работа начинается с функции f , которая отображает состояние мира W на стимулы S , создаваемые этим миром:

$$S = f(W)$$

Функция f определена в физике и в оптике, а также достаточно хорошо изучена. Задача выработки стимулов S с использованием функции f и данных о реальном или воображаемом мире W решается в области **компьютерной графики**. Задача машинного зрения в определенном смысле является обратной задачей компьютерной графики — в ней предпринимается попытка вычислить W с помощью f и S по следующей формуле:

$$W = f^{-1}(S)$$

К сожалению, функция f не имеет приемлемой обратной функции. Прежде всего, мы не можем заглянуть за угол, поэтому не имеем возможности восстановить все аспекты мира из полученных зрительных стимулов. Более того, даже наблюдаемая часть мира представляется как чрезвычайно неоднозначная — без дополнительной информации нельзя сказать, содержит ли стимул S изображение игрушечного ящера Годзилла, ломающего шестидесятисантиметровый макет здания, или в нем представлен настоящий монстр, разрушающий здание высотой шестьдесят метров. Некоторые из подобных проблем можно решить, составив распределение вероятностей по мирам, а не пытаясь найти уникальный мир:

$$P(W) = P(W|S) P(S)$$

Еще более важным недостатком моделирования такого типа является предпринимаемая в нем попытка решить слишком трудную проблему. Достаточно сказать, что в компьютерной графике может потребоваться несколько часов вычислений для того, чтобы прорисовать единственный кадр кинофильма, притом что в секунду требуется 24 таких кадра; к тому же вычисление функции f^{-1} гораздо сложнее по сравнению с вычислением f . Очевидно, что такой объем вычислений слишком велик даже для суперкомпьютера, не говоря уже об обычной мухе, если требуется обеспечить реагирование в реальном времени. К счастью, агенту не требуется модель с таким уровнем детализации, который используется в компьютерной графике, где нужно добиться, чтобы построенное изображение стало таким же реальным, как настоящая фотография. Агенту достаточно знать, скрывается ли в кустарнике тигр, а не учитывать все данные о точном местонахождении и ориентации каждого волоска на спине этого тигра.

Основная часть настоящей главы посвящена описанию средств, позволяющих обеспечить распознавание объектов, таких как притаившиеся тигры, и в ней будут показаны способы решения этой задачи без представления данных о самом тигре до мельчайших подробностей. В разделе 24.2 описан процесс формирования изображения и определены некоторые особенности функции $f(W)$. Вначале рассматривается геометрия этого процесса. Будет описано, как свет отражается от объектов во внешнем мире и попадает на точки в плоскости изображения оптического датчика искусственного агента. Геометрия объясняет, почему большой ящер Годзилла, находящийся далеко от нас, кажется таким же, как маленький ящер Годзилла, расположенный намного ближе. После этого рассматривается фотометрия данного процесса, что позволяет понять, как свет в наблюдаемой сцене определяет яркость точек на изображении. Геометрия и фотометрия, вместе взятые, позволяют получить модель того, как объекты во внешнем мире отображаются на двумерный массив пикселей.

Получив представление о том, как формируются изображения, мы перейдем к изучению способов их обработки. Процесс обработки потока визуальной информации как людьми, так и компьютерами можно разделить на три этапа. На раннем этапе обработки, называемом *зрением низкого уровня* (раздел 24.3), необработанное изображение сглаживается для устранения шума и извлекаются характеристики двумерного изображения, в частности данные о краях участков, разделяющих регионы изображения. На этапе визуальной обработки среднего уровня эти края группируются в целях формирования двумерных областей. А на этапе обеспечения зрения высокого уровня (раздел 24.4) эти двумерные области распознаются как действительные объекты в реальном мире (раздел 24.5). Мы изучим различные элементы изображения, позволяющие более успешно решать эту задачу, включая признаки движения, стереоданные, текстуру, затенение и контуры. Задача распознавания объектов важна для агентов, действующих в условиях дикой природы, чтобы они могли обнаруживать присутствие тигров, а также важна для промышленных роботов, чтобы они могли отличать гайки от болтов. Наконец, в разделе 24.6 описано, как можно использовать результаты распознавания объектов для выполнения полезных задач, таких как манипулирование объектами и навигация. Средства манипулирования позволяют захватывать и использовать инструменты и другие объекты, а средства навигации дают возможность передвигаться из одного места в другое без столкновений с какими-либо препятствиями. Не упуская из виду эти задачи, можно добиться того, чтобы агент формировал модель только в таком объеме, который позволяет ему успешно достичь своих целей.

24.2. ФОРМИРОВАНИЕ ИЗОБРАЖЕНИЯ

В процессе зрения концентрируется свет, рассеянный объектами в *сцене*, и создается двумерное *изображение* на плоскости изображения. Плоскость изображения покрыта светочувствительным материалом — в сетчатке таковым являются молекулы родопсина, на фотографической пленке — галогены серебра, а в цифровой камере — массив элементов с зарядовой связью (Charge-Coupled Device — CCD). Каждый элемент в приборе с зарядовой связью (ПЗС) накапливает заряд, пропорциональный количеству электронов, освобожденных в результате поглощения фотонов за фиксированный период времени. В цифровой камере плоскость

изображения представлена в виде прямоугольной решетки, состоящей из нескольких миллионов **пикселей**. В глазу имеется аналогичный массив элементов, состоящий примерно из 100 миллионов палочек и 5 миллионов колбочек, сгруппированных в гексагональный массив.

Сцена очень велика, а плоскость изображения весьма мала, поэтому требуется определенный способ фокусировки света на плоскости изображения. Такая операция может быть выполнена с помощью линзы или без нее. В любом случае наша основная задача состоит в том, чтобы определить геометрию происходящих преобразований и обеспечить возможность прогнозировать, где каждая точка сцены найдет свое представление на плоскости изображения.

Получение изображения без линз — камера-обскура

Простейший способ формирования изображения состоит в использовании **камеры-обскуры**, в конструкцию которой входит микроотверстие O в передней части ящика и плоскость изображения в задней части ящика (рис. 24.1). Мы будем использовать трехмерную систему координат с началом координат в точке O и рассматривать точку P в сцене, имеющую координаты (X, Y, Z) . Точка P проектируется в точку P' на плоскости изображения с координатами (x, y, z) . Если f — расстояние от микроотверстия до плоскости изображения, то с помощью теоремы подобия треугольников можно получить следующие уравнения:

$$\frac{-x}{f} = \frac{X}{Z}, \quad \frac{-y}{f} = \frac{Y}{Z} \Rightarrow x = \frac{-fX}{Z}, \quad y = \frac{-fY}{Z}$$

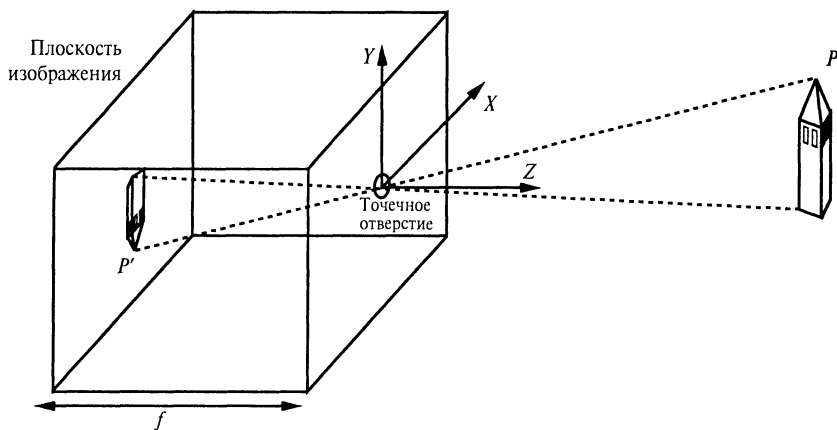


Рис. 24.1. Геометрия формирования изображения в камере-обскуре

Эти уравнения определяют процесс формирования изображения, называемый **перспективной проекцией**. Заслуживает внимания то, что значение координаты Z находится в знаменателе, а это означает, что чем дальше объект, тем меньше его изображение. Кроме того, наличие знака “минус” означает, что изображение инвертировано, т.е. повернуто на 180° по сравнению с самой сценой.

При перспективной проекции параллельные линии сходятся в одной точке на горизонте (достаточно представить себе уходящие вдаль железнодорожные рельсы).

Рассмотрим, почему так должно быть. Линия в сцене, проходящая через точку (X_0, Y_0, Z_0) в направлении (U, V, W) , может быть описана как множество точек $(X_0 + \lambda U, Y_0 + \lambda V, Z_0 + \lambda W)$, где λ изменяется в пределах от $-\infty$ до $+\infty$. Проекция точки P_λ от этой линии до плоскости изображения задается следующей формулой:

$$\left(f \frac{X_0 + \lambda U}{Z_0 + \lambda W}, f \frac{Y_0 + \lambda V}{Z_0 + \lambda W} \right)$$

По мере того как $\lambda \rightarrow \infty$ или $\lambda \rightarrow -\infty$, эта формула принимает вид $P_\infty = (fU/W, fV/W)$, если $W \neq 0$. Точку P_∞ называют **точкой схода**, связанной с семейством прямых линий с ориентацией (U, V, W) . Все линии, имеющие одну и ту же ориентацию, имеют и одинаковую точку схода.

Если объект имеет относительно небольшую глубину по сравнению с его расстоянием от камеры, появляется возможность аппроксимировать перспективную проекцию с помощью **масштабированной ортогональной проекции**. Идея такой операции состоит в следующем: если глубина Z точек объекта изменяется в некоторых пределах $Z_0 \pm \Delta Z$, где $\Delta Z \ll Z_0$, то коэффициент перспективного масштабирования f/Z можно приближенно представить с помощью константы $s = f/Z_0$. Уравнения для проекции, которые связывают координаты сцены (X, Y, Z) с координатами плоскости изображения, принимают вид $x = sX$ и $y = sY$. Следует отметить, что масштабированная ортогональная проекция представляет собой аппроксимацию, действительную только для таких частей сцены, которые не характеризуются значительными изменениями внутренней глубины; эта проекция должна использоваться только для исследования свойств “в малом”, а не “в большом”. В качестве примера, позволяющего убедиться в необходимости соблюдать осторожность, отметим, что при использовании ортогональной проекции параллельные линии остаются параллельными, а не сливаются в точке схода!

Системы линз

В глазах позвоночных и в современных видеокамерах используются **линзы**. Линза имеет гораздо большую площадь по сравнению с микроотверстием, что позволяет пропускать с ее помощью больше света. За это приходится платить тем, что исчезает возможность представить в резком фокусе всю сцену одновременно. Изображение объекта в сцене, находящегося на расстоянии Z , создается на фиксированном расстоянии от линзы Z' , а отношение между Z и Z' задается с помощью следующего уравнения линзы, где f — фокусное расстояние линзы:

$$\frac{1}{Z} + \frac{1}{Z'} = \frac{1}{f}$$

Если дана определенная возможность выбора расстояния изображения Z_0' между узловой точкой линзы и плоскостью изображения, то точки сцена с глубинами в диапазоне, близком к Z_0 , где Z_0 — соответствующее расстояние до объекта, могут быть спроектированы на изображение в достаточно резком фокусе. Указанный диапазон глубин в сцене называется **глубиной резкости пространственного изображения**.

Следует отметить, что расстояние до объекта Z обычно намного больше по сравнению с расстоянием до изображения Z' или по сравнению с f , поэтому часто можно воспользоваться следующей аппроксимацией:

$$\frac{1}{Z} + \frac{1}{Z'} \approx \frac{1}{Z'} \Rightarrow \frac{1}{Z'} \approx \frac{1}{f}$$

Таким образом, расстояние до изображения $Z' \approx f$. Поэтому можно по-прежнему использовать уравнения перспективной проекции камеры-обскуры для описания геометрии формирования изображения в системе линз.

Для того чтобы можно было создавать сфокусированные изображения объектов, находящихся на разных расстояниях Z , линза в глазу (рис. 24.2) меняет форму, а линза в камере передвигается в направлении Z .

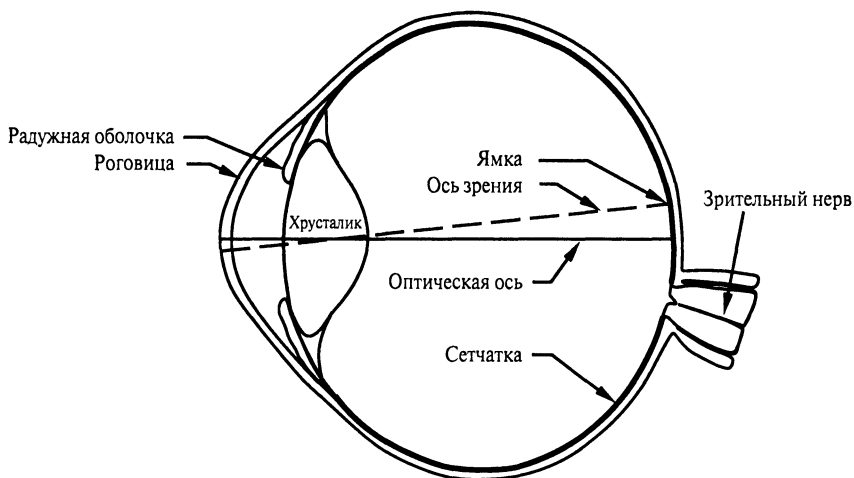


Рис. 24.2. Горизонтальный поперечный разрез человеческого глаза

Свет: фотометрия формирования изображения

Свет — это наиболее важная предпосылка зрения; без света все изображения были бы одинаково темными, независимо от того, насколько интересной является сцена. **Фотометрия** — это наука о свете. Для наших целей мы создадим модель того, как свет в сцене преобразуется в интенсивность света на плоскости изображения, которая обозначается¹ как $I(x, y)$. Такая модель лежит в основе любой системы зрения, позволяющей выявлять по данным об интенсивности света на изображениях свойства внешнего мира. На рис. 24.3 показано оцифрованное изображение степлера на столе, а также отмечен квадратом блок пикселей с размерами 12×12 , выделенный из изображения степлера. Работа любой компьютерной программы, предназначенной для интерпретации изображения, начинается с матрицы значений эффективности, подобной этой.

Яркость пиксела на изображении пропорциональна количеству света, направленного в камеру от конечной части поверхности, ограниченной замкнутой кривой, в сцене, которая проектируется на данный пиксел. Это значение, в свою очередь, зависит от отражательных свойств рассматриваемой конечной части поверхности, а также от положения и распределения источников света в сцене. Кроме того, отражательные

¹ Если требуется также учитывать изменения интенсивности во времени, то используется выражение $I(x, y, t)$.

свойства зависят от остальной части сцены, поскольку другие поверхности в сцене могут стать косвенными источниками света, отражая падающий на них свет.

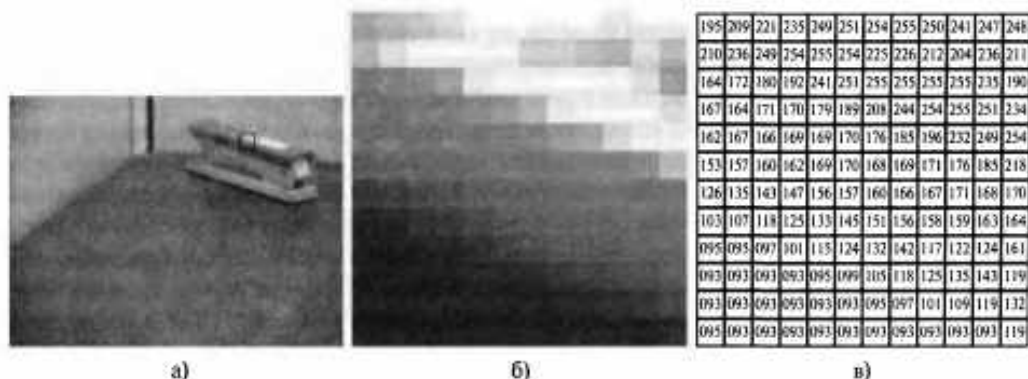


Рис. 24.3. Пример исходных данных для анализа изображения: фотография степлера на столе (а); блок пикселей размерами 12×12, взятый из изображения на рис. 24.3, а, в увеличенном виде (б); соответствующие значения яркости изображения, изменяющиеся в пределах от 0 до 255 (в)

Мы можем моделировать два различных вида отражения. **Зеркальное отражение** означает, что свет отражается от внешней поверхности объекта и подчиняется тому закону, что угол падения равен углу отражения. Так действует идеальное зеркало. **Диффузное отражение** означает, что свет проникает через поверхность объекта, поглощается объектом, а затем снова излучается за пределы его поверхности. Если поверхность является идеально рассеивающей (или **ламбертовой**), то свет рассеивается с равной интенсивностью во всех направлениях. Интенсивность зависит только от угла падения луча, поступающего от источника света: если источник света расположен прямо над поверхностью, то отражается больше всего света, а если лучи от источника света проходят почти параллельно поверхности, то количество отраженного света примерно равно нулю. Между этими двумя крайностями интенсивность отражения I подчиняется закону косинуса, установленному Ламбертом:

$$I = k I_0 \cos \theta$$

где I_0 — интенсивность источника света; θ — угол между источником света и перпендикуляром к поверхности; k — константа, называемая **отражательной способностью**, которая зависит от отражательных свойств поверхности. Она изменяется от 0 (для совершенно черных поверхностей) до 1 (для чисто белых поверхностей).

В действительности почти все поверхности обладают сочетанием свойств диффузного и зеркального отражения. Моделирование такого сочетания на компьютере — это квинтэссенция компьютерной графики. Прорисовка реалистических изображений обычно осуществляется по методу трассировки луча, цель которого заключается в моделировании физического процесса генерации света источниками света, а затем многократного повторного отражения.

Цвет — спектрофотометрия формирования изображения

На рис. 24.3 мы представили черно-белое изображение, полностью игнорируя тот факт, что видимый свет складывается из целого ряда волн с разной длиной, начиная

от 400 нанометров (нм) на фиолетовом и заканчивая 700 нм на красном конце спектра. В некоторых случаях свет состоит из волн, имеющих лишь единственное значение длины, соответствующее одному из цветов радуги. Но в других случаях свет представляет собой комбинацию волн различной длины. Означает ли это, что в качестве меры для описанной выше величины $I(x, y)$ необходимо использовать сочетание значений, а не единственное значение? Если бы нам требовалось точно представить физические свойства света, то действительно возникла бы указанная выше необходимость. Но если нам нужно лишь эмулировать процесс восприятия света людьми (и многими другими позвоночными), то можно пойти на компромисс. Эксперименты (начатые еще Томасом Юнгом в 1801 году) показали, что любая смесь световых волн с разными значениями длины, вне зависимости от ее сложности, может быть представлена в виде смеси, состоящей лишь из трех основных цветов. Это означает, что если есть генератор света, позволяющий составлять линейные комбинации световых волн с тремя значениями длины (как правило, для этого выбирают красный (700 нм), зеленый (546 нм) и синий (436 нм)), то путем регулировки рукояток для увеличения относительного содержания одного цвета и уменьшения другого можно составить любую комбинацию значений длин волн; по крайней мере, если полученная комбинация предназначена для восприятия человеком. Этот экспериментальный факт означает, что изображения могут быть представлены с помощью вектора, определяющего только три значения интенсивности в расчете на один пиксел, и каждое из этих значений должно соответствовать интенсивностям света с тремя основными значениями длины волны. На практике для воспроизведения изображения с высоким качеством достаточно отвести по одному байту для каждого значения. Правильность такого подхода к обеспечению трехцветного восприятия цвета подтверждается также тем, что в сетчатке имеются три типа колбочек, пиковое значение чувствительности которых находится в диапазоне значений длины волны соответственно 650, 530 и 430 нм. Однако возникающие при этом связи намного сложнее, чем можно было бы представить с помощью взаимно-однозначного отображения.

24.3. ОПЕРАЦИИ, ВЫПОЛНЯЕМЫЕ НА ПЕРВОМ ЭТАПЕ ОБРАБОТКИ ИЗОБРАЖЕНИЯ

Как было указано выше, свет, отражаясь от объектов в сцене, формирует изображение, состоящее, скажем, из пяти миллионов трехбайтовых пикселов. Как и при использовании датчиков всех других типов, полученный сигнал содержит шум, но в данном случае ситуация усугубляется тем, что объем полученных данных очень велик. В этом разделе будет показано, что можно сделать с данными изображения для того, чтобы упростить их обработку. Вначале рассмотрим операции сглаживания изображения, позволяющие уменьшить шум, а также операции, позволяющие обнаруживать края участков на изображении. Эти операции называются операциями “предварительной обработки” или операциями “низкого уровня”, поскольку они стоят на первом месте в конвейере операций. Визуальные операции предварительной обработки характеризуются тем, что выполняются локально (они могут применяться лишь к одному участку изображения без учета того, что есть еще какие-то другие участки изображения, пусть даже находящиеся на расстоянии всего нескольких пикселов), а также тем, что в них не требуются знания:

для сглаживания изображения и обнаружения краев не нужно задумываться над тем, какие объекты представлены на изображениях. Благодаря этому операции низкого уровня вполне могут быть реализованы с помощью параллельных обрабатывающих средств либо в живом организме, либо в электронном устройстве. Затем мы рассмотрим одну операцию среднего уровня — операцию сегментации изображения на участки. Операции на этом этапе все еще применяются к изображению, а не ко всей сцене, но уже появляются элементы нелокальной обработки.

Как было указано в разделе 15.2, под **сглаживанием** подразумевается предсказание значения переменной состояния в некоторый момент времени t в прошлом при наличии свидетельств, полученных, начиная с момента времени t и заканчивая всеми другими значениями времени вплоть до настоящего момента. Теперь мы применим такую же идею, но не во временной проблемной области, а в пространственной, и будем трактовать процесс сглаживания как предсказание значения данного конкретного пиксела, если известны значения окружающих его пикселей. Следует отметить, что необходимо четко понимать, каково различие между наблюдаемым значением, измеренным для какого-то пиксела, и истинным значением, которое в действительности должно было быть измерено в этом пикселе. Они могут быть разными из-за случайных ошибок измерений или из-за систематического отказа, поскольку может оказаться, что в этой точке неисправен элемент матрицы CCD.

Один из способов сглаживания изображения состоит в том, чтобы каждому пикселу присваивалось среднее значение характеристик его соседних пикселей. Такой способ обработки, как правило, исключает экстремальные значения. Но остается открытым вопрос: сколько нужно рассмотреть соседних пикселей — один, два или больше? Ответ на этот вопрос заключается в том, что для исключения гауссова шума следует рассчитать взвешенное среднее с использованием **фильтра с гауссовой характеристикой**. Напомним, что гауссова функция со среднеквадратичным отклонением σ выражается следующими формулами:

$$G_{\sigma}(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-x^2/2\sigma^2} \quad \text{в одном измерении или}$$

$$G_{\sigma}(x, y) = \frac{1}{\sqrt{2\pi}\sigma} e^{-(x^2+y^2)/2\sigma^2} \quad \text{в двух измерениях}$$

Под применением фильтра с гауссовой характеристикой подразумевается замена значения интенсивности $I(x_0, y_0)$ суммой по всем (x, y) пикселям значений $I(x, y) G_{\sigma}(d)$, где d — расстояние от (x_0, y_0) до (x, y) . Такого рода взвешенная сумма применяется так часто, что для нее предусмотрено особое название и обозначение. Считается, что функция h представляет собой **свертку** двух функций, f и g (обозначается как $h = f * g$), если имеют место следующие соотношения:

$$h(x) = \sum_{u=-\infty}^{+\infty} f(u) g(x-u) \quad \text{в одном измерении или}$$

$$h(x, y) = \sum_{u=-\infty}^{+\infty} \sum_{v=-\infty}^{+\infty} f(u, v) g(x-u, y-v) \quad \text{в двух измерениях}$$

Поэтому операция сглаживания осуществляется путем формирования свертки изображения и гауссова распределения, $I * G_\sigma$. Значение σ , равное 1 пикселу, является достаточным для сглаживания шума с небольшой интенсивностью. Если же значение σ соответствует 2 пикселам, то происходит сглаживание шума с большей интенсивностью, но теряются некоторые мелкие детали. Поскольку влияние гауссова распределения уменьшается с расстоянием, на практике можно заменить пределы $\pm\infty$ в суммах значениями, примерно равными $\pm 3\sigma$.

Обнаружение краев

Следующая операция, выполняемая на первом этапе обработки изображения, состоит в обнаружении краев на плоскости изображения. **Краями** называются прямые или кривые линии на плоскости изображения, которые служат "водоразделом" для существенных изменений в яркости изображения. Целью обнаружения краев является повышение уровня абстракции и переход от перегруженного подробностями мультимегабайтового изображения к более компактному, абстрактному представлению, как показано на рис. 24.4. Необходимость в выполнении такой операции обусловлена тем, что линии краев на изображении соответствуют важным контурам в сцене. На этом рисунке приведены три примера изображений, на которых отмечены так называемые *сосредоточенные неоднородности*: сосредоточенная неоднородность по глубине, обозначенная меткой 1; сосредоточенные неоднородности по ориентации двух поверхностей, обозначенные меткой 2; сосредоточенная неоднородность по коэффициенту отражения, обозначенная меткой 3; сосредоточенная неоднородность по освещенности (тень), обозначенная меткой 4. Результаты операции обнаружения краев относятся только к данному конкретному изображению и поэтому не содержат данных, позволяющих распознать эти различные типы сосредоточенных неоднородностей в сцене, но такая задача может быть решена в ходе дальнейшей обработки.

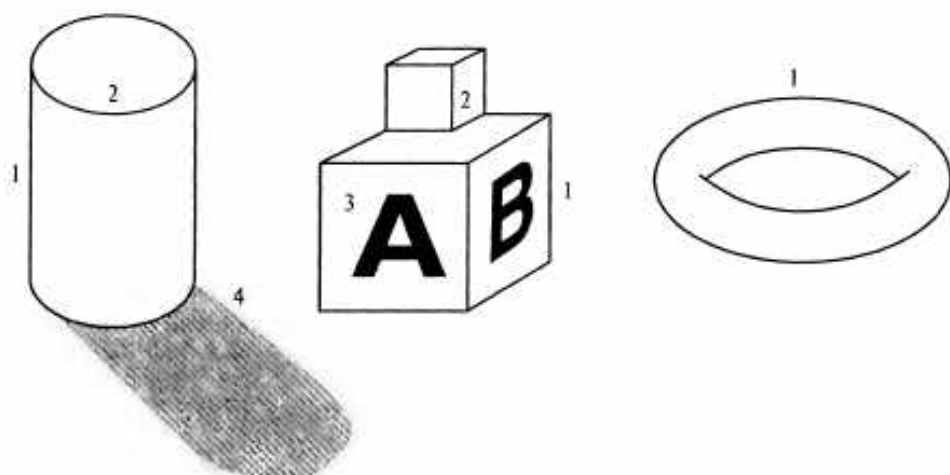
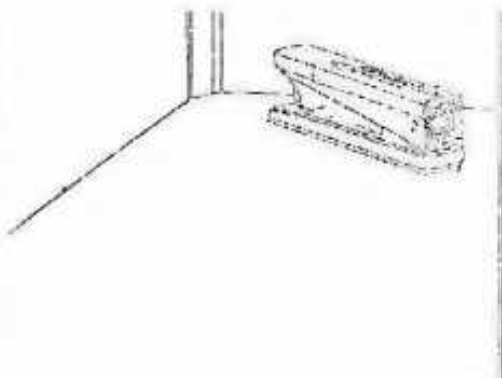


Рис. 24.4. Различные виды краев: сосредоточенные неоднородности по глубине (1); сосредоточенные неоднородности по ориентации поверхностей (2); сосредоточенные неоднородности по коэффициенту отражения (3); сосредоточенные неоднородности по освещенности (тени) (4)

На рис. 24.5, *а* приведено изображение сцены, на котором показан степлер, находящийся на столе, а на рис. 24.5, *б* приведены результаты применения алгоритма обнаружения краев к этому изображению. Вполне очевидно, что эти результаты весьма отличаются от идеального контурного рисунка. Небольшие края, соответствующие деталям изображения, не всегда выровнены друг с другом; на некоторых участках, где эти края обрываются, имеются пропуски, кроме того, есть и края, появившиеся под воздействием шума, которые не соответствуют каким-либо значимым деталям в этой сцене. Все указанные ошибки должны быть исправлены на последующих этапах обработки. Теперь перейдем к описанию того, как происходит обнаружение краев на изображении. Рассмотрим профиль яркости изображения вдоль одномерного поперечного разреза, перпендикулярного к некоторому краю (например, между левой стороной стола и стеной). Он выглядит примерно так, как показано на рис. 24.6, *а*. Местонахождение края соответствует значению $x=50$.



а)



б)

Рис. 24.5. Пример исходного изображения и результатов его обработки: фотография степлера (*а*); края, обнаруженные в результате обработки изображения на рис. 24.5, *а* (*б*)

Поскольку края соответствуют тем участкам изображения, где яркость подвергается резким изменениям, то на первый взгляд может показаться, что достаточно дифференцировать изображение и найти такие места, где производная $I'(x)$ принимает большое значение. И действительно, такой подход в определенной степени осуществим. Тем не менее, как показано на рис. 24.6, *б*, кроме основного пикового значения с координатой $x=50$, обнаруживаются также дополнительные пиковые значения в других местах (например, с координатой $x=75$), которые могут быть ошибочно приняты за настоящие края. Эти дополнительные пиковые значения возникают из-за наличия шума в изображении. Если вначале будет проведено сглаживание изображения, то количество фиктивных пиков уменьшится, как показано на рис. 24.6, *в*.

Теперь у нас появляется возможность применить на данном этапе определенную оптимизацию, поскольку операции сглаживания и поиска краев можно объединить в одну операцию. Существует теорема, согласно которой для любых функций f и g производная свертки, $(f * g)'$, равна свертке с производной, $f * (g)'$. Поэтому вместо сглаживания изображения с последующим дифференцированием можно просто выполнить свертку изображения с производной гауссовой функции сглаживания,

G_{σ}' . Таким образом, алгоритм поиска края в одном измерении может быть представлен следующим образом.

1. Выполнить свертку изображения I с функцией G_{σ}' для получения результатов свертки R .

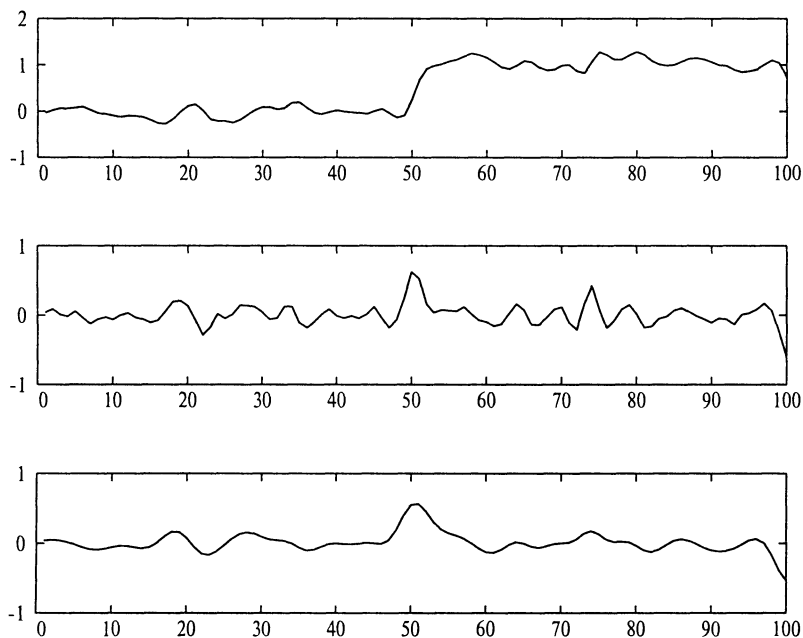


Рис. 24.6. Верхний рисунок: профиль интенсивности $I(x)$ вдоль одномерного разреза, пересекающего край, который характеризуется ступенчатым изменением яркости. Средний рисунок: производная интенсивности, $I'(x)$. Большие значения на этом графике соответствуют краям, но представленные на нем данные содержат шум. Нижний рисунок: производная от сглаженной версии данных об интенсивности, $(I * G_{\sigma})'$, которая может быть вычислена в одном шаге как свертка $I * G_{\sigma}'$. Исчез появившийся под воздействием шума пик с координатой $x=75$, который в других условиях рассматривался бы как признак наличия края

2. Обозначить как края те пиковые значения в $||R(x)||$, которые превышают некоторое заранее заданное пороговое значение T . Это пороговое значение выбирается таким образом, чтобы можно было устранить фиктивные пиковые значения, возникшие под воздействием шума.

В двух измерениях края могут проходить под любым углом θ . Для обнаружения вертикальных краев можно применить такую очевидную стратегию: выполнить свертку с $G_{\sigma}'(x)G_{\sigma}(y)$. В направлении y эффект этой операции сведется к тому, что будет выполнено только сглаживание (под воздействием гауссовой свертки), а в направлении x результатом операции станет то, что дифференцирование будет сопровождаться сглаживанием. Поэтому алгоритм обнаружения вертикальных краев состоит в следующем.

1. Выполнить свертку изображения $I(x, y)$ с функцией $f_v(x, y) = G_\sigma'(x) G_\sigma(y)$ для получения результатов свертки $R_v(x, y)$.
2. Обозначить как края те пиковые значения в $||R_v(x, y)||$, которые превышают некоторое заранее заданное пороговое значение T .

Для того чтобы обнаружить какой-либо край, имеющий произвольную ориентацию, необходимо выполнить свертку изображения с двумя фильтрами, $f_v = G_\sigma'(x) G_\sigma(y)$ и $f_h = G_\sigma'(y) G_\sigma(x)$, где функция f_h соответствует функции f_v , график которой повернут на 90° . Таким образом, алгоритм обнаружения краев, имеющих произвольную ориентацию, состоит в следующем.

1. Выполнить свертку изображения $I(x, y)$ с функциями $f_v(x, y)$ и $f_h(x, y)$ для получения соответственно результатов свертки $R_v(x, y)$ и $R_h(x, y)$. Определить $R(x, y) = R_v^2(x, y) + R_h^2(x, y)$.
2. Обозначить как края те пиковые значения в $||R(x, y)||$, которые превышают некоторое заранее заданное пороговое значение T .

После того как с помощью этого алгоритма будут отмечены пиксели краев, на следующем этапе необходимо связать друг с другом те пиксели, которые принадлежат к одним и тем же кривым краев. Такую операцию можно выполнить, приняв предположение, что любые два соседних пиксела, которые являются пикселями края с совместимыми ориентациями, должны принадлежать к одной и той же кривой края. Описанный выше процесс получил название процесса **обнаружения края Кэнни** в честь его разработчика Джона Кэнни (John Canny).

Края после их обнаружения становятся основой для многих этапов последующей обработки: их можно использовать для выполнения стереооптической обработки, обнаружения движения или распознавания объектов.

Сегментация изображения

Мозг человека не использует полученные им результаты восприятия в непосредственном виде, а организует эти результаты определенным образом, поэтому вместо коллекции значений яркости, связанных с отдельными фоторецепторами, мозг выделяет целый ряд визуальных групп, которые обычно ассоциируются с объектами или частями объектов. Такая способность является не менее важной и для машинного зрения.

Сегментация — это процесс разбиения изображения на группы с учетом подобия характеристик пикселей. Основная идея этого процесса состоит в следующем: каждый пиксел изображения может быть связан с некоторыми визуальными свойствами, такими как яркость, цвет и текстура². В пределах одного объекта или одной части объекта эти атрибуты изменяются относительно мало, тогда как при переходе через границу от одного объекта к другому обычно происходит существенное изменение одного или другого из этих атрибутов. Необходимо найти вариант разбиения изображения на такие множества пикселей, что указанные ограничения удовлетворяются в максимально возможной степени.

² Анализ свойств текстуры базируется на статистических данных, полученных применительно к небольшому замкнутому участку поверхности, центром которого является рассматриваемый пиксел.

Существует целый ряд различных способов, с помощью которых эта интуитивная догадка может быть формализована в виде математической теории. Например, в [1402] рассматриваемая задача представлена как задача сегментации графа. Узлы графа соответствуют пикселям, а ребра — соединениям между пикселями. Ребрам, соединяющим пары пикселей i и j , присваиваются веса W_{ij} с учетом того, насколько близки значения яркости, цвета, текстуры и т.д. для двух пикселей соответствующей пары. Затем осуществляется поиск разбиений, которые минимизируют нормализованный критерий отсечения. Грубо говоря, критерием сегментации графа является критерий минимизации суммы весов соединений между группами и максимизации суммы весов соединений в пределах групп.

Процесс сегментации, основанный исключительно на использовании низкоуровневых локальных атрибутов, таких как яркость и цвет, чреват существенными ошибками. Чтобы надежно обнаруживать границы, связанные с объектами, необходимо также использовать высокоуровневые знания о том, какого рода объекты могут по всей вероятности встретиться в данной сцене. При распознавании речи такая возможность появилась благодаря использованию формальных средств скрытой марковской модели; в контексте обработки изображений поиск такой универсальной инфраструктуры остается темой интенсивных исследований. Так или иначе представление высокоуровневых знаний об объектах является темой следующего раздела.

24.4. ИЗВЛЕЧЕНИЕ ТРЕХМЕРНОЙ ИНФОРМАЦИИ

В данном разделе будет показано, как перейти от двухмерного изображения к трехмерному представлению сцены. Для нас важно перейти именно к стилю рассуждений о сцене в связи с тем, что агент в конечном итоге существует в мире, а не на плоскости изображения, а зрение предназначено для получения возможности взаимодействовать с объектами в том мире, где существует агент. Тем не менее для большинства агентов требуется только ограниченное абстрактное представление некоторых аспектов сцены, а не все подробности. Алгоритмы, используемые при решении задач взаимодействия с окружающим миром, которые были приведены в последних частях данной книги, распространяются на краткие описания объектов, а не на исчерпывающие перечисления каждой трехмерной конечной части поверхности, ограниченной замкнутой кривой.

Вначале в этом разделе рассматривается процесс **распознавания объекта**, в котором характеристики изображения (такие как края) преобразуются в модели известных объектов (таких как степлеры). Распознавание объекта происходит в три этапа: сегментация сцены с выделением различных объектов, определение позиции и ориентации каждого объекта относительно наблюдателя и определение формы каждого объекта.

Определение позиции и ориентации объекта относительно наблюдателя (так называемой **позы объекта**) является наиболее важной операцией с точки зрения решения задач манипулирования и навигации. Например, чтобы робот мог передвигаться по полу заводского цеха в условиях ограниченного маневра, он должен знать местонахождение всех препятствий, чтобы иметь возможность спланировать путь, позволяющий избежать столкновения с ними. Если же робот должен выбрать и захватить какой-то объект, то он должен знать расположение этого объекта относи-

тельно манипулятора, чтобы выработать подходящую траекторию движения. Действия по манипулированию и навигации обычно осуществляются в рамках заданного контура управления, а сенсорная информация предоставляет обратную связь для модификации движения робота или перемещения манипулятора робота.

Представим позицию и ориентацию в математических терминах. Позиция точки P в сцене характеризуется тремя числами — координатами (X, Y, Z) точки P в системе координат с началом координат в микроотверстии и с осью Z , проходящей вдоль оптической оси (см. рис. 24.1). В нашем распоряжении имеется перспективная проекция точки на изображении (x, y) . Эта проекция определяет луч, проходящий из микроотверстия, на котором расположена точка P ; неизвестным является расстояние. Термин “ориентация” может использоваться в двух описанных ниже смыслах.

1. Ориентация объекта как единого целого. Она может быть задана в терминах трехмерного вращения, связывающего систему координат этого объекта с системой координат камеры-обскуры.
2. Ориентация поверхности объекта в точке P . Она может быть задана с помощью нормального вектора \mathbf{n} , т.е. вектора, задающего направление, перпендикулярное к поверхности. Для представления ориентации поверхности часто используются переменные \sphericalangle **угол поворота** и \sphericalangle **угол наклона**. *Углом поворота* называется угол между осью Z и вектором \mathbf{n} , а *углом наклона* — угол между осью X и проекцией вектора \mathbf{n} на плоскость изображения.

По мере перемещения камеры-обскуры по отношению к объекту изменяются и расстояние до объекта, и его ориентация. Сохраняется только \sphericalangle **форма** объекта. Если объект представляет собой куб, он остается таковым и после его перемещения. В геометрии попытки формализовать понятие геометрической формы предпринимались в течение многих столетий; в конечном итоге было сформулировано такое основное понятие, что формой является то, что остается неизменным после применения некоторой группы преобразований, например сочетаний поворотов и переносов. Сложность заключается в том, что нужно найти способ представления глобальной формы, достаточно общий для того, чтобы с его помощью можно было описать широкий перечень объектов реального мира (а не только такие простые формы, как цилиндры, конусы и сферы) и при этом предусмотреть возможность легко восстанавливать информацию о форме из визуальных входных данных. Но гораздо лучше изучена проблема описания локальной формы поверхности. По сути, это может быть выполнено в терминах кривизны — определения того, как изменяется положение нормального вектора по мере передвижения в различных направлениях по этой поверхности. Если поверхность представляет собой плоскость, то положение нормального вектора вообще не изменяется. В случае цилиндрической поверхности при перемещении параллельно оси изменения не происходят, а при перемещении в перпендикулярном направлении вектор, нормальный к поверхности, вращается со скоростью, обратно пропорциональной радиусу цилиндра, и т.д. Все эти явления исследуются в научной области, называемой *дифференциальной геометрией*.

Форму объекта важно знать при выполнении некоторых задач манипулирования (например, чтобы определить, в каком месте лучше всего захватить данный объект), но наиболее значительную роль форма объекта играет при распознавании объектов. При решении последней задачи наиболее значащими подсказками, позволяющими

идентифицировать объекты, определять с помощью методов классификации, что это изображение является примером какого-то класса, встречавшегося ранее, и т.д., служат геометрическая форма наряду с цветом и текстурой.

Фундаментальный вопрос состоит в следующем: “Если дано, что во время создания перспективной проекции все точки в трехмерном мире, расположенные вдоль одного луча, проходящего через микроотверстие, спроектировались на одну и ту же точку в изображении, то как теперь восстановить трехмерную информацию?” В визуальных стимулах, относящихся к числу применимых для этой цели, содержится целый ряд характерных признаков, включая **движение**, **бинокулярные стереоданные**, **текстуру**, **затенение** и **контуры**. Каждый из этих характерных признаков позволяет опереться на исходные предположения о физических сценах, чтобы можно было получить (почти) полностью непротиворечивые интерпретации этих сцен. Каждый из указанных характерных признаков рассматривается в пяти приведенных ниже разделах.

Движение

До сих пор рассматривалось только одно изображение одновременно. Но видеокамеры позволяют получать 30 кадров в секунду и различия между кадрами могут стать важным источником информации. Если видеокамера движется относительно трехмерной сцены, то в изображении возникает кажущееся результирующее движение, называемое **оптическим потоком**. Оптический поток содержит информацию о направлении и скорости движения характеристик в изображении, являющегося результатом относительного движения между наблюдателем и сценой. На рис. 24.7, а, б показаны два кадра из видеофильма о вращении кубика Рубика. На рис. 24.7, в показаны векторы оптического потока, вычисленные на основании этих двух изображений. В оптическом потоке зашифрована полезная информация о структуре сцены. Например, при наблюдении из движущегося автомобиля удаленные объекты характеризуются гораздо более медленным кажущимся движением по сравнению с близкими объектами, поэтому скорость кажущегося движения позволяет получить определенную информацию о расстоянии.

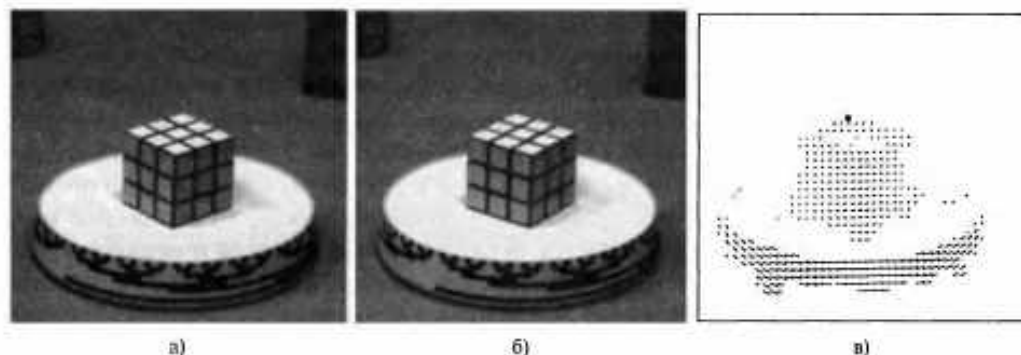


Рис. 24.7. Пример использования понятия оптического потока: кубик Рубика на поворотном столе, приведенном во вращение (а); тот же кубик, показанный через 19/30 секунды (любезно предоставлено Ричардом Шелиски (Richard Szeliski)) (б); векторы потока, вычисленные путем сравнения двух изображений, приведенных на рис. 24.7, а, б (любезно предоставлено Джо Вебером (Joe Weber) и Джитендрой Маликом (Jitendra Malik)) (в)

Поле вектора оптического потока может быть представлено с помощью его компонентов $v_x(x, y)$ в направлении x и $v_y(x, y)$ в направлении y . Для измерения оптического потока необходимо найти соответствующие точки между одним временным кадром и следующим. При этом используется тот факт, что замкнутые участки изображения, сосредоточенные вокруг соответствующих точек, характеризуются аналогичными шаблонами интенсивности. Рассмотрим блок пикселей с центром в пикселе p , в точке (x_0, y_0) , во время t_0 . Этот блок пикселей необходимо сравнить с блоками пикселей, центрами которых являются различные потенциально применимые пиксели q_i с координатами $(x_0 + D_x, y_0 + D_y)$ во время $t_0 + D_t$. Одним из возможных критериев подобия является **сумма квадратов разностей** (Sum of Squared Differences — SSD):

$$SSD(D_x, D_y) = \sum_{(x, y)} (I(x, y, t) - I(x + D_x, y + D_y, t + D_t))^2$$

Здесь координаты (x, y) принимают свои значения среди пикселей в блоке с центром в точке (x_0, y_0) . Найдем значения (D_x, D_y) , которые минимизируют выражение для SSD. В таком случае оптический поток в точке (x_0, y_0) принимает значение $(v_x, v_y) = (D_x / D_t, D_y / D_t)$. Еще один вариант состоит в том, что можно максимизировать **взаимную корреляцию** следующим образом:

$$Correlation(D_x, D_y) = \sum_{(x, y)} I(x, y, t) I(x + D_x, y + D_y, t + D_t)$$

Метод с использованием взаимной корреляции действует лучше всего, если сцена характеризуется наличием текстуры, в результате чего блоки пикселей (называемые также *окнами*) содержат значительные вариации яркости среди входящих в них пикселей. Если же рассматривается ровная белая стена, то взаимная корреляция обычно остается почти одинаковой для различных потенциальных согласований q и алгоритм сводится к операции выдвигения слепого предположения.

Допустим, что наблюдатель движется с линейной скоростью (или скоростью переноса) T и с угловой скоростью ω (таким образом, эти параметры описывают **самодвижение**). Можно вывести уравнение, связывающее скорости наблюдателя, оптический поток и положения объектов в сцене. Если предположить, что $f=1$, то из этого следуют уравнения

$$\begin{aligned} v_x(x, y) &= \left[-\frac{T_x}{Z(x, y)} - \omega_y + \omega_z y \right] - x \left[-\frac{T_z}{Z(x, y)} - \omega_x y + \omega_y x \right] \\ v_y(x, y) &= \left[-\frac{T_y}{Z(x, y)} - \omega_z x + \omega_x y \right] - y \left[-\frac{T_z}{Z(x, y)} - \omega_x y + \omega_y x \right] \end{aligned}$$

где $Z(x, y)$ задает координату z точки в сцене, соответствующей точке на изображении с координатами (x, y) .

Достаточно хорошего понимания того, что при этом происходит, можно достичь, рассмотрев случай чистого переноса. В таком случае выражения для поля потока принимают следующий вид:

$$v_x(x, y) = \frac{-T_x + x T_z}{Z(x, y)}, \quad v_y(x, y) = \frac{-T_y + y T_z}{Z(x, y)}$$

Теперь становятся очевидными некоторые интересные свойства. Оба компонента оптического потока, $v_x(x, y)$ и $v_y(x, y)$, принимают нулевое значение в точке с координатами $x = T_x / T_z$, $y = T_y / T_z$. Эта точка называется **фокусом расширения** поля потока. Предположим, что мы изменим начало координат в плоскости x – y для того, чтобы оно находилось в фокусе расширения; в таком случае выражение для оптического потока принимает особенно простую форму. Допустим, что (x', y') — это новые координаты, определяемые соотношениями $x' = x - T_x / T_z$, $y' = y - T_y / T_z$. В таком случае становятся справедливыми следующие уравнения:

$$v_x(x', y') = \frac{x' T_z}{Z(x', y')}, \quad v_y(x', y') = \frac{y' T_z}{Z(x', y')}$$

Эти уравнения имеют некоторые интересные области применения. Предположим, что летящая муха пытается сесть на стену и хочет определить, в какой момент она коснется стены, если будет сохраняться текущая скорость. Это время задается термом Z / T_z . Следует отметить, что мгновенное значение поля оптического потока не позволяет получить ни данные о расстоянии Z , ни данные о компоненте скорости T_z , но вместе с тем предоставляет значение соотношения этих двух параметров и поэтому может использоваться для управления приближением к поверхности посадки. Эксперименты, проведенные над мухами в реальных условиях, показали, что указанные насекомые действуют именно по этому принципу. Мухи относятся к числу наиболее ловких летающих объектов среди всех живых существ и машин, а интересное всего то, что они добиваются этого с помощью системы зрения, имеющей невероятно низкое пространственное разрешение (поскольку в глазу мухи имеется всего около 600 рецепторов, тогда как в глазу человека — порядка 100 миллионов), но блестящее временное разрешение.

Чтобы получить данные о глубине, необходимо воспользоваться несколькими кадрами. Если видеокамера направлена на твердое тело, то его форма не изменяется от кадра к кадру и поэтому появляется возможность лучше решать задачу измерения оптического потока, неизменно подверженного шуму. Результаты применения одного из подобных подходов, полученные Томази и Канаде [1511], показаны на рис. 24.8 и 24.9.

Бинокулярные стереоданные

Большинство позвоночных имеют два глаза. Это не только удобно в качестве резерва на случай потери одного глаза, но и дает также некоторые другие преимущества. У многих травоядных глаза расположены по обе стороны головы, что позволяет им иметь более широкий обзор. С другой стороны, у хищников глаза находятся в передней части головы, благодаря чему они используют **бинокулярные стереоданные**. Принцип бинокулярного зрения аналогичен принципу, в котором используется параллакс, возникающий во время движения, за тем исключением, что вместо использования изображений, сменяющихся во времени, применяются два изображения (или, в случае систем машинного зрения, и большее количество изображений), которые разделены в пространстве; например, такие изображения передают в мозг глаза человека, смотрящего прямо вперед. Поскольку данная конкретная характеристика в сцене будет находиться в различных местах по отношению к оси z каждой плоскости изображения, то после совмещения двух изображений будет обнаруживаться

✎ **рассогласование** в местонахождениях этой характеристики в двух изображениях. Такое явление можно наблюдать на рис. 24.10, где ближайшая точка пирамиды сдвинута влево на правом изображении и вправо на левом изображении.

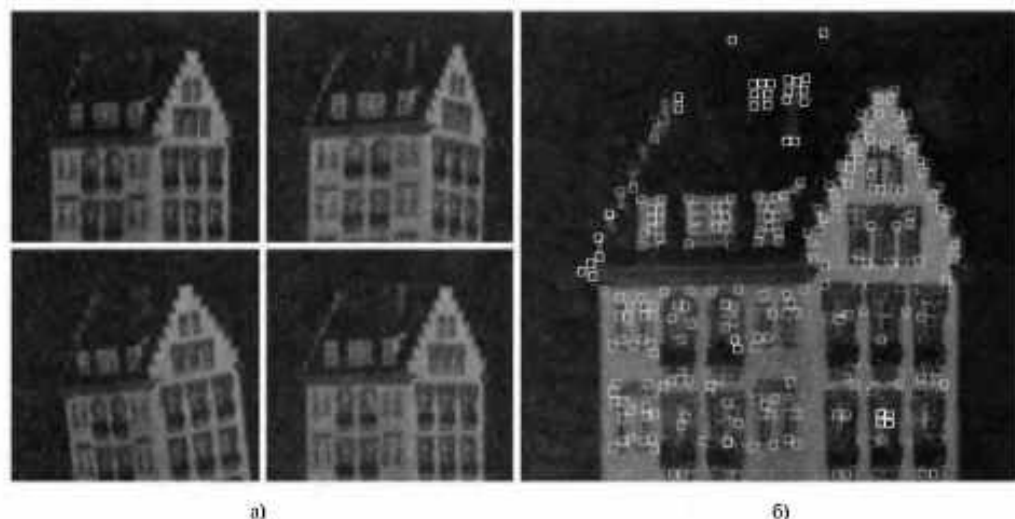


Рис. 24.8. Пример применения нескольких кадров: четыре кадра из видеопоследовательности, в которой камера передвигается и вращается относительно объекта (а); первый кадр из последовательности, обозначенный небольшими прямоугольниками, подчеркивающими характерные особенности, которые были обнаружены детектором характеристик (любезно представлено Карлом Томази (Carlo Tomasi)) (б)

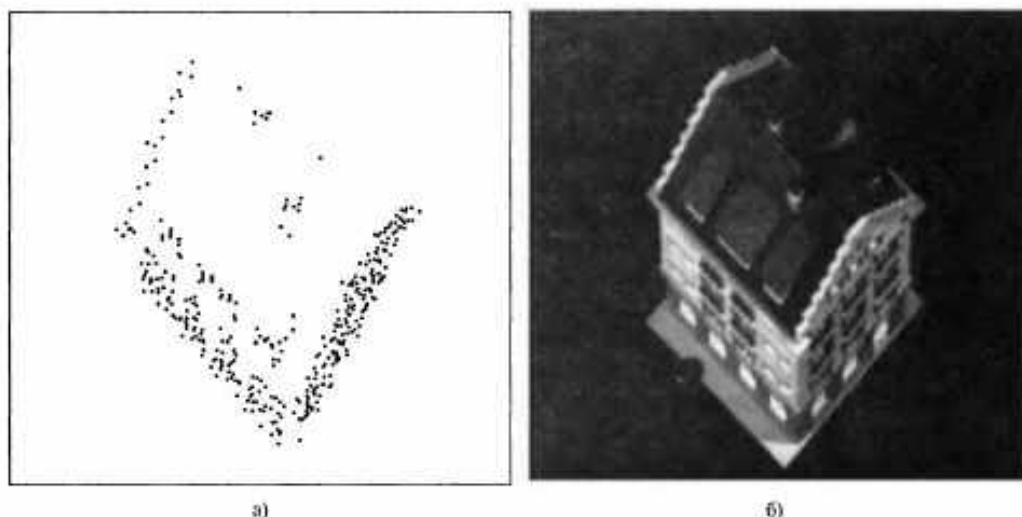


Рис. 24.9. Пример результатов обработки нескольких кадров: трехмерная реконструкция местонахождений характеристик изображения, показанных на рис. 24.8 (вид сверху) (а); настоящий дом, снимок которого получен из той же позиции (б)

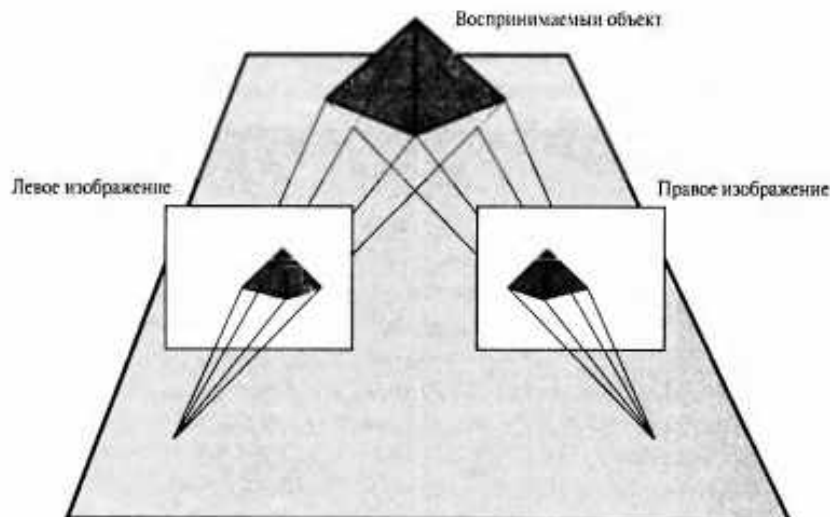


Рис. 24.10. Принцип формирования стереоданных: изменение положения видеокамеры приводит к получению немного отличающихся друг от друга двухмерных представлений одной и той же трехмерной сцены

Проанализируем геометрические соотношения между рассогласованием и глубиной. Прежде всего рассмотрим случай, в котором оба глаза (или обе видеокамеры) направлены прямо вперед, поэтому их оптические оси параллельны. В таком случае связь между изображением в правой и в левой видеокамере состоит в том, что для преобразования одного изображения в другое достаточно выполнить сдвиг вдоль оси x на величину b (расстояние между камерами, или опорная линия). Для вычисления рассогласований по горизонтали и вертикали, которые выражаются в виде $H = v_x \Delta t$, $V = v_y \Delta t$, можно использовать уравнения оптического потока из предыдущего раздела, при условии, что $T_x = b / \Delta t$ и $T_y = T_z = 0$. Параметры вращательного сдвига ω_x , ω_y и ω_z равны нулю. Таким образом, $H = b / Z$, $V = 0$. Иными словами, рассогласование по горизонтали равно отношению длины опорной линии к глубине, а рассогласование по вертикали равно нулю.

При нормальных условиях наблюдения люди **фиксируют** свой взгляд; это означает, что они выбирают в качестве объекта наблюдения некоторую точку в сцене, и в этой точке пересекаются оптические оси двух глаз. На рис. 24.11 показана ситуация, в которой взгляд человека, смотрящего двумя глазами, зафиксирован в точке P_0 , находящейся на расстоянии Z от средней точки между глазами. Для удобства мы будем вычислять угловое рассогласование, измеряемое в радианах. Рассогласование в точке фиксации P_0 равно нулю. Для некоторой другой точки P в сцене, которая находится дальше на величину δZ , можно вычислить угловые смещения левого и правого изображений точки P , которые будут обозначаться соответственно P_L и P_R . Если каждое из этих изображений смещено на угол $\delta\theta/2$ относительно P_0 , то смещение между P_L и P_R , которое представляет собой рассогласование в точке P , выражается величиной $\delta\theta$. Простые геометрические преобразования позволяют получить следующее выражение:

$$\frac{\delta\theta}{\delta Z} = \frac{-b}{Z^2}$$

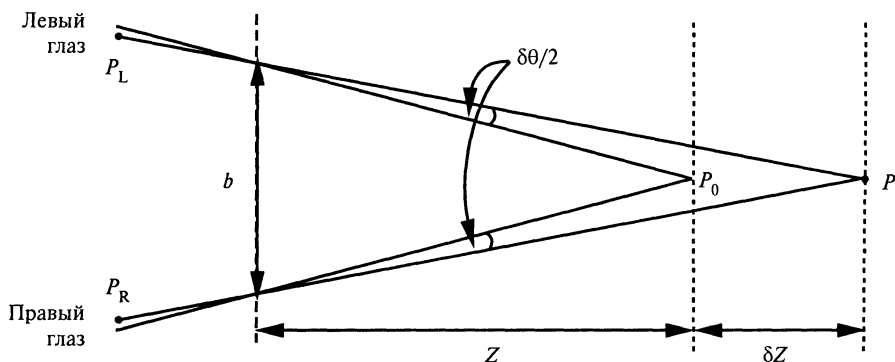


Рис. 24.11. Зависимость между рассогласованием и глубиной в стереоданных

У людей величина b (\approx базисная линия) составляет около 6 см. Предположим, что расстояние Z примерно равно 100 см. В таком случае наименьшее доступное обнаружению значение $\delta\theta$ (соответствующее размеру светочувствительного элемента глаза) составляет около 5 угловых секунд, а это означает, что разность расстояний δZ равна 0,4 мм. При $Z=30$ см получим значение $\delta Z=0,036$ мм, которое на удивление мало. Это означает, что на расстоянии 30 см люди способны различать значения глубины, которые отличаются друг от друга на столь малую величину, как 0,036 мм, что позволяет им продевать нитку через ушко иглки и выполнять другие тонкие операции.

Градиенты текстуры

В повседневной речи под \approx **текстурой** подразумевается свойство поверхностей, связанное с осязательными ощущениями, которое оно напоминает (слово “текстура” имеет тот же корень, что и слово “текстиль”). А в проблематике машинного зрения этим словом обозначается тесно связанное понятие, определяющее наличие повторяющегося в пространстве рисунка на поверхности, который может быть обнаружен визуально. В качестве примеров можно назвать одинаковый ряд окон на здании, стежки на свитере, пятна на шкуре леопарда, травинки на лужайке, гальку на берегу и толпу людей на стадионе. Иногда взаимное расположение повторяющихся рисунков является весьма регулярным, как в случае стежков на свитере, а в других случаях, таких как галька на берегу, регулярность обеспечивается только в определенном статистическом смысле: плотность расположения гальки приблизительно одинакова в разных частях пляжа.

Сказанное выше является справедливым применительно только к реальной сцене, а на изображении кажущиеся размеры, форма, взаиморасположение и другие характеристики элементов текстуры (называемых \approx **текселями**) действительно различаются, как показано на рис. 24.12. Например, черепицы выглядят одинаковыми только в реальной сцене, а на изображении размеры и форма проекций черепиц варьируют, и это связано с двумя описанными ниже основными причинами.

1. Различия в расстояниях от текселов до видеокамеры. Напомним, что в перспективной проекции удаленные объекты кажутся меньшими. Коэффициент масштабирования равен $1/Z$.

2. Различия в ракурсах текстелов. Эта причина связана с ориентацией каждого текстела относительно линии зрения, направленной от камеры. Если текстел расположен перпендикулярно линии зрения, то ракурс отсутствует. Величина эффекта ракурса пропорциональна $\cos\sigma$, где σ — угол поворота плоскости текстела.

Проведя определенные выкладки в рамках математического анализа, можно вычислить выражения для скорости изменения различных характеристик текстелов изображения, таких как площадь, ракурс и плотность. Такие показатели называются **градиентами текстуры** и являются функциями от формы поверхности, а также от углов ее поворота и наклона по отношению к местонахождению наблюдателя.

Для восстановления данных о форме из данных о текстуре можно использовать следующий двухэтапный процесс: а) измерить градиенты текстуры; б) определить оценочные значения формы поверхности, а также углов ее поворота и наклона, которые могли бы привести к получению измеренных градиентов текстуры. Результаты применения этого процесса приведены на рис. 24.12.



а)



б)

Рис. 24.12. Примеры применения процесса восстановления формы: сцена, на которой показан градиент текстуры. Восстановление данных об ориентации поверхности может быть выполнено на основании предположения, что реальная текстура является однородной. Вычисленное значение ориентации поверхности показано путем наложения на изображение белого кружка и указателя, трансформированного так, как если бы кружок был нарисован на поверхности в этом месте (а); восстановление данных о форме по данным о текстуре в случае криволинейной поверхности (изображения любезно предоставлены Джитендрой Маликом (Jitendra Malik) и Рут Розенхолц (Ruth Rosenholtz) [972]) (б)

Затенение

Затенение (под этим подразумевается изменение интенсивности света, полученного от различных участков поверхности в сцене) определяется геометрией сцены и отражательными свойствами поверхностей. В компьютерной графике создание затенения сводится к вычислению значений яркости изображения $I(x, y)$ с учетом геометрии сцены и отражательных свойств объектов в сцене. В проблематике машинного зрения решается обратная задача — восстановление данных о геометрии и отражательных свойствах по данным об яркости изображения $I(x, y)$. Как оказалось, эта задача с трудом поддается решению, за исключением самых простейших случаев.

Начнем с ситуации, в которой действительно можно найти решение задачи определения данных о форме на основании данных о затенении. Рассмотрим ламбертову поверхность, свет на которую падает от удаленного точечного источника света. Предположим, что поверхность находится достаточно далеко от видеокамеры, чтобы можно было использовать ортогональную проекцию в качестве аппроксимации перспективной проекции. Яркость изображения определяется с помощью следующей формулы:

$$I(x, y) = k \mathbf{n}(x, y) \cdot \mathbf{s}$$

где k — константа масштабирования; \mathbf{n} — единичный вектор, нормальный к поверхности; \mathbf{s} — единичный вектор, направленный в сторону источника света. Поскольку \mathbf{n} и \mathbf{s} — единичные векторы, их точечное произведение представляет собой косинус угла между ними. Форму поверхности можно определить, следя за тем, как изменяется направление нормального вектора \mathbf{n} , движущегося вдоль поверхности. Предположим, что значения k и \mathbf{s} известны. Поэтому задача сводится к тому, чтобы восстановить данные о векторе $\mathbf{n}(x, y)$, нормальном к поверхности, если известна интенсивность изображения $I(x, y)$.

Прежде всего необходимо отметить, что задача определения \mathbf{n} , если дана яркость I в данном пикселе (x, y) , не разрешима локально. Может быть вычислен угол, под которым вектор \mathbf{n} пересекается с вектором, направленным к источнику света, но полученный результат позволяет лишь узнать, что этот вектор находится в определенном конусе направлений с осью \mathbf{s} и углом от вершины $\theta = \cos^{-1}(I/k)$. Чтобы перейти к более точным вычислениям, необходимо отметить, что вектор \mathbf{n} не может изменяться произвольно при переходе от пиксела к пикселу. Он соответствует нормальному вектору гладкой конечной части поверхности, ограниченной замкнутой кривой, и поэтому также должен изменяться плавно (формальным названием для этого ограничения является **интегрируемость**). На основе этой идеи было разработано несколько различных методов. Один из них состоит в том, что нужно использовать другое выражение для \mathbf{n} , в терминах частных производных Z_x и Z_y глубины $Z(x, y)$. Такой подход приводит к получению частного дифференциального уравнения для Z , которое может быть решено для получения данных о глубине $Z(x, y)$ с учетом подходящих граничных условий.

Этот подход может быть немного обобщен. Например, не обязательно, чтобы поверхность была ламбертовой, а источник света был точечным. При условии, что существует возможность вычислить **карту коэффициентов отражения** $R(\mathbf{n})$, которая задает значения яркости конечной части поверхности, ограниченной замкнутой кривой, как функции от положения нормального вектора \mathbf{n} к этой поверхности, могут по сути применяться методы такого же типа.

Настоящие сложности возникают, когда приходится иметь дело со взаимными отражениями. Если рассматривается типичная сцена внутри помещения, такая как сцена, в которой показаны объекты внутри офиса, то можно заметить, что поверхности освещаются не только источниками света, но и светом, отраженным от других поверхностей в сцене, которые фактически служат в качестве вторичных источников света. Такие эффекты взаимного освещения являются весьма значительными. В подобной ситуации формальный подход, предусматривающий использование карты коэффициентов отражения, становится полностью неприменимым, поскольку яр-

кость изображения зависит не только от положения нормального вектора к поверхности, но также и от сложных пространственных связей между различными поверхностями в сцене.

Не подлежит сомнению, что люди обладают определенными способностями к восприятию данных о форме на основании данных о затенении, поэтому указанная задача продолжает привлекать значительный интерес, несмотря на все сложности, связанные с ее решением.

Контуры

Рассматривая контурный рисунок, подобный приведенному на рис. 24.13, мы получаем живое восприятие трехмерной формы и расположения. Благодаря чему это происходит? Ведь уже было сказано выше, что к получению одного и того же контурного рисунка приводит обработка не одной конфигурации сцены, а бесконечного количества таких конфигураций. Кроме того, следует отметить, что контурный рисунок позволяет даже получить представление о наклоне и повороте поверхностей. Такое ощущение может достигаться благодаря использованию сочетания знаний высокого уровня (знаний о типичных формах) с ограничениями низкого уровня.

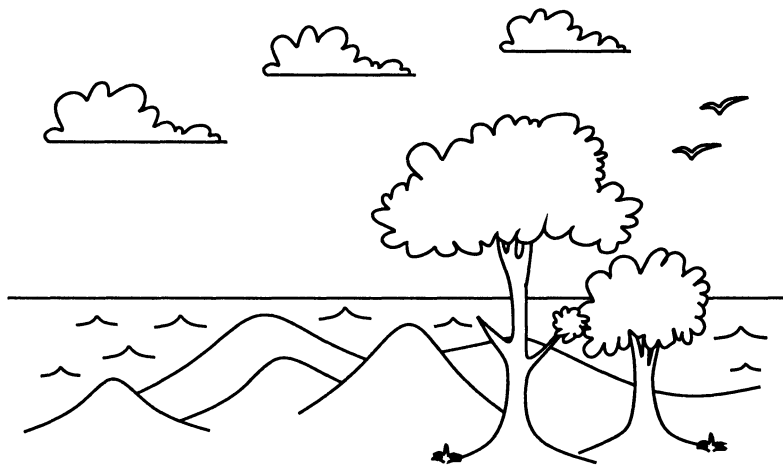


Рис. 24.13. Контурный рисунок, позволяющий получить полное представление о том, что на нем изображено (любезно предоставлен Айшей Маликом (Isha Malik))

Рассмотрим, какие качественные знания могут быть получены с помощью контурного рисунка. Как было описано выше, линии на рисунке могут иметь много разных трактовок (см. рис. 24.4 и его подрисовочную подпись). Задача оценки фактической значимости каждой линии на изображении называется **разметкой линий**; она была одной из первых задач, изучаемых в области машинного зрения. На данный момент займемся изучением упрощенной модели мира, в которой объекты не имеют отметок на поверхности, а линии, обусловленные наличием сосредоточенных неоднородностей освещенности, такие как края теней и блики, были удалены на каком-то из этапов предварительной обработки, что позволяет нам сконцентрировать

свое внимание на контурных рисунках, где каждая линия соответствует сосредоточенной неоднородности либо по глубине, либо по ориентации.

В таком случае каждую линию можно отнести к одному из двух классов: рассматривать ее как проекцию **лимба** (геометрического места тех точек на поверхности, где луч зрения проходит по касательной к поверхности) или как **край** (поверхностная нормальная сосредоточенная неоднородность). Кроме того, каждый край может быть классифицирован как выпуклый, вогнутый или закрывающий (под этим подразумевается, что он закрывает то, что находится за ним). Что касается закрывающих краев и лимбов, то желательно иметь возможность определять, какая из двух поверхностей, примыкающих к кривой на контурном рисунке, является ближайшей к наблюдателю в данной сцене. Такие наложения линий могут быть представлены путем присваивания каждой линии одной из шести перечисленных ниже возможных **меток линий**, как показано на рис. 24.14.

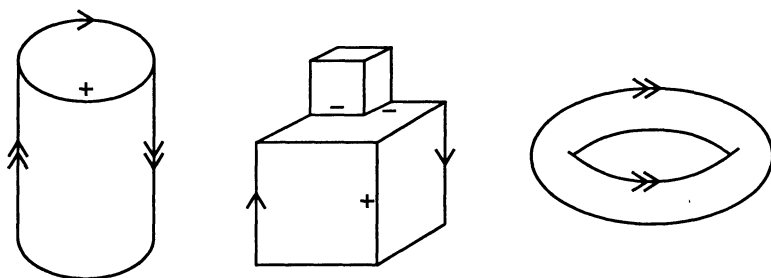


Рис. 24.14. Различные виды меток линий

1. Метки + и - представляют соответственно выпуклые и вогнутые края. Они связаны с поверхностными нормальными сосредоточенными неоднородностями, в которых видны обе поверхности, стыкующиеся вдоль этого края.
2. Метка \leftarrow или \rightarrow представляет закрывающий выпуклый край. При просмотре сцены из видеокамеры обе конечные части поверхности, ограниченные замкнутой кривой, которые стыкуются вдоль этого края, лежат на одной и той же стороне, но одна из них закрывает другую. По мере перемещения по направлению стрелки эти закрывающие поверхности остаются справа.
3. Метка $\leftarrow\leftarrow$ или $\rightarrow\rightarrow$ представляет лимб. На этой линии поверхность плавно искривляется по кругу, закрывая саму себя. По мере перемещения в направлении, обозначенном двойной стрелкой, закрывающая поверхность остается справа. Луч зрения проходит по касательной к поверхности во всех точках лимба. По мере изменения точки зрения лимбы меняют свое положение на поверхности объекта.

Если количество линий на рисунке равно n , то количество вариантов присваивания меток линий, определяемое законами комбинаторики, равно 6^n , но количество физически возможных вариантов присваивания по сравнению с этим количеством составляет лишь очень небольшую величину. Задача определения таких возможных присваиваний меток называется *задачей разметки линий*. Следует отметить, что эта задача имеет смысл, только если метка остается одинаковой на всем протяжении линии. Но такое условие не всегда соблюдается, поскольку метки могут изменяться

вдоль линий на изображениях выпукло-вогнутых криволинейных объектов. В настоящей главе для предотвращения указанных сложностей будут рассматриваться исключительно только многогранные объекты.

Хаффмен [702] и Клоувс [271] независимо друг от друга впервые предприняли попытку применить систематический подход к анализу сцен с многогранными объектами. В своем анализе Хаффмен и Клоувс ограничивались сценами с непрозрачными **трехгранными** твердыми телами; таковыми являются объекты, в которых в каждой вершине сходятся три и только три плоские поверхности. В случае наличия сцен с многочисленными объектами они, кроме этого, исключали такие выравнивания объектов, которые могли бы привести к нарушению предположения о наличии только трехгранных объектов, например сцен, в которых два куба имеют общий край. Не допускалось также наличие **трешин** (т.е. "краев", вдоль которых касательные плоскости являются непрерывными). Для такого мира трехгранных объектов Хаффмен и Клоувс подготовили исчерпывающий список всех различных типов вершин и описали всевозможные способы, с помощью которых эти вершины могут рассматриваться под общей точкой зрения. Условие, согласно которому должна существовать общая точка зрения, фактически гарантирует то, что если возникает небольшое движение глаза наблюдателя, ни одно из соединений плоскостей не меняет свой характер. Например, из этого условия следует, что если три линии пересекаются на изображении, то должны также пересекаться соответствующие края в сцене.

Четыре способа, с помощью которых три плоские поверхности могут быть соединены в одной вершине, показаны на рис. 24.15. Эти примеры могут быть также составлены путем деления куба на восемь **октантов**. В таком случае различные возможные трехгранные вершины в центре куба создаются путем заполнения разных октантов. Вершина, обозначенная цифрой 1, соответствует одному заполненному октанту, вершина с цифрой 3 — трем заполненным октантам и т.д. Рекомендуем читателям самим убедиться в том, что на данном рисунке действительно представлены все возможности. Например, попытка заполнить два октанта в кубе не приводит к созданию допустимой трехгранной вершины в центре. Следует также отметить, что эти четыре случая соответствуют различным комбинациям выпуклых и вогнутых краев, которые встречаются в данной вершине.

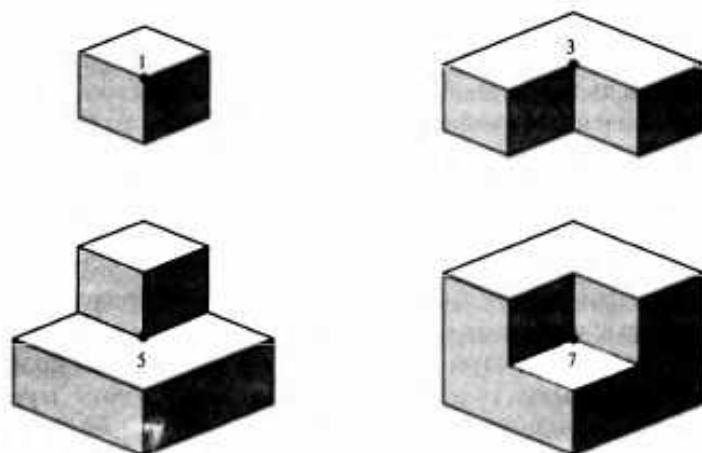


Рис. 24.15. Четыре вида трехгранных вершин

Три края, встречающихся в вершине, делят окружающее пространство на восемь октантов. Вершина видна из любого октанта, не заполненного твердым материалом. Перемещение точки зрения в пределах одного октанта не приводит к получению изображения с различными типами соединений. Вершина, обозначенная цифрой 1 на рис. 24.15, может рассматриваться из любого из оставшихся семи октантов; при этом наблюдаются метки соединения, показанные на рис. 24.16.

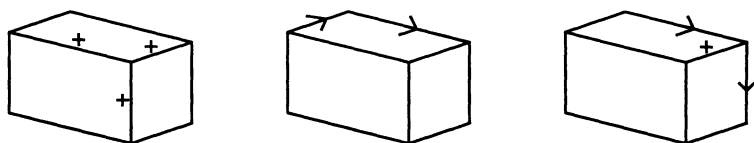


Рис. 24.16. Изменение внешнего вида вершины, обозначенной цифрой 1 на рис. 24.15

Работа по составлению исчерпывающего списка различных способов, с помощью которых может рассматриваться каждая вершина, привела к получению вариантов, показанных на рис. 24.17. Получено четыре различных типа соединений, которые могут быть выделены на изображении: L-соединения, Y-соединения, стреловидные соединения и Т-соединения. L-соединения соответствуют двум видимым краям. Y-соединения и стреловидные соединения соответствуют результатам рассмотрения трех краев, но различие между ними состоит в том, что в Y-соединении ни один из трех углов не превышает 180° . Т-соединения связаны с закрытием одной поверхности другой. Если ближайшая, непрозрачная поверхность закрывает вид на дальше расположенный ее край, будет получен непрерывный край, который встречается с частично закрытым краем. Четыре метки Т-соединения соответствуют закрытию четырех различных типов краев.

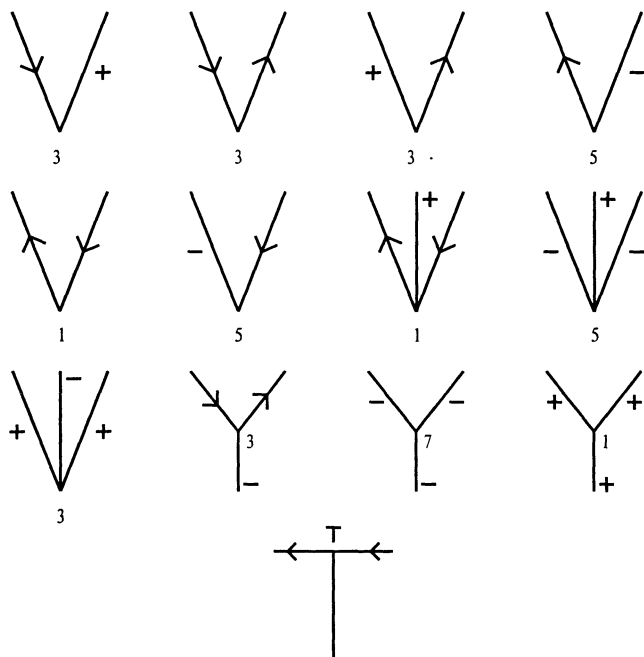


Рис. 24.17. Множество меток Хаффмена–Клуэвса

При использовании этого словаря соединений во время поиска разметки для контурного рисунка приходится решать задачу определения того, какие интерпретации соединений являются глобально совместимыми. Соблюдение свойства совместимости обеспечивается путем применения правила, согласно которому каждой линии на рисунке вдоль всей ее длины должна быть присвоена одна и только одна метка. Вальц [1552] предложил алгоритм решения этой задачи (фактически применимый даже для расширенной ее версии с тенями, трещинами и разделимо вогнутыми краями), который стал одним из первых приложений метода удовлетворения ограничений в искусственном интеллекте (см. главу 5). В терминологии задач CSP переменными являются соединения, значениями — разметки для этих соединений, а ограничениями служит то, что каждая линия имеет единственную метку. Хотя задача разметки линии для сцен с трехгранными объектами является NP-полной, на практике стандартные алгоритмы CSP показали высокую производительность при их решении.

24.5. РАСПОЗНАВАНИЕ ОБЪЕКТОВ

Зрение позволяет нам надежно распознавать людей, животных и неодушевленные объекты. В области искусственного интеллекта или машинного зрения для обозначения всех этих способностей принято использовать термин *распознавание объектов*. К этому относится определение класса конкретных объектов, представленных на изображении (например, лица), а также распознавание самих конкретных объектов (например, лица Билла Клинтона). Ниже перечислены прикладные области, которые стимулируют развитие этого научно-технического направления.

- **❧ Биометрическая идентификация.** Криминальные расследования и контроль доступа на объекты, допускающие присутствие ограниченного круга лиц, требуют наличия возможности однозначно идентифицировать личность людей. Операции снятия отпечатков пальцев, сканирования радужной оболочки глаза и фотографирования лица в фас приводят к получению изображений, которые должны быть сопоставлены с данными, относящимися к конкретным людям.
- **❧ Выборка изображений с учетом их содержимого.** В текстовом документе можно легко найти местонахождение любой строки, например “cat” (кошка), если она там имеется; такую возможность предоставляет любой текстовый редактор. А теперь рассмотрим задачу поиска в изображении подмножества пикселей, которые соответствуют изображению кошки. Если бы система машинного зрения обладала такой способностью, то позволяла бы отвечать на запросы, касающиеся содержимого изображений, такие как “Найдите фотографию, на которой показаны вместе Билл Клинтон и Нельсон Мандела”, “Найдите фотографию конькобежца, который в процессе бега полностью оторвался ото льда”, “Найдите фотографию Эйфелевой башни ночью” и т.д., без необходимости вводить ключевые слова, озаглавливающие каждую фотографию в коллекции. По мере того как увеличиваются коллекции фотографий и видеофильмов, задача ввода вручную аннотаций к отдельным объектам из этой коллекции становится все сложнее.
- **❧ Распознавание рукописного текста.** К примерам такого текста относятся подписи, блоки адресов на конвертах, суммы в чеках и введенные пером данные в персональных цифровых ассистентах (Personal Digital Assistant — PDA).

Зрение используется для распознавания не только объектов, но и видов деятельности. Люди способны узнавать знакомую походку (издалека замечая своего друга), выражение лица (улыбку, гримасу), жест (например, просьбу приблизиться), действие (прыжок, танец) и т.д. Исследования по распознаванию видов деятельности все еще находятся на этапе своего становления, поэтому в данном разделе мы сосредоточимся на теме распознавания объектов.

Люди, как правило, легко решают задачу распознавания объектов, но практика показала, что эта задача является сложной для компьютеров. Дело в том, что система машинного зрения должна обладать способностью идентифицировать лицо человека, несмотря на изменения освещенности, позы по отношению к видеокамере и выражения лица. Любое из этих изменений вызывает появление широкого перечня различий в значениях яркости пикселей, поэтому метод, предусматривающий простое сравнение пикселей, вряд ли окажется применимым. Если же требуется обеспечить распознавание экземпляров определенной категории, такой как “автомобили”, то приходится также учитывать различия внутри самой категории. Как оказалось, значительные трудности возникают даже при попытке решить весьма ограниченную проблему распознавания рукописных цифр в поле для почтового кода на конвертах.

Наиболее подходящую инфраструктуру для изучения проблемы распознавания объектов предоставляют такие научные области, как контролируемое обучение или классификация образов. Системе предъявляют положительные примеры изображений (допустим, “лица” — *face*) и отрицательные примеры (допустим, “не лица” — *nonface*) и ставят перед ней задачу определить с помощью обучения функцию, которая позволила бы отнести вновь полученные изображения к одной из двух категорий — *face*, *nonface*. Для достижения этой цели подходят все методы, описанные в главах 18 и 20; в частности, для решения проблем распознавания объектов были применены многослойные перцептроны, деревья решений, классификаторы по ближайшим соседним элементам и ядерные машины. Но следует отметить, что задача приспособить эти методы для распознавания объектов — далеко не такая уж простая.

Прежде всего необходимо преодолеть сложности, связанные с сегментацией изображения. Любое изображение, как правило, содержит множество объектов, поэтому необходимо вначале разбить его на подмножества пикселей, соответствующих отдельным объектам. А после разбиения изображения на участки можно ввести данные об этих участках или совокупностях участков в классификатор для определения меток объектов. К сожалению, процесс сегментации “снизу вверх” чреват ошибками, поэтому в качестве альтернативного подхода может быть предусмотрен поиск для определения групп объектов “сверху вниз”. Это означает, что можно проводить поиск подмножества пикселей, которые можно классифицировать как лицо, и в случае успешного выполнения данного этапа результатом становится успешное обнаружение группы! Но подходы, основанные исключительно на поиске “сверху вниз” (или нисходящем поиске), имеют высокую вычислительную сложность, поскольку в них необходимо исследовать окна изображения различных размеров, находящиеся в разных местах, а также сравнивать их все с данными различных гипотез о наличии объектов. В настоящее время такая нисходящая стратегия используется в большинстве практически применяемых систем распознавания объектов, но подобная ситуация может измениться в результате усовершенствования методов поиска “снизу вверх” (восходящего поиска).

Еще одной причиной затруднений является то, что процесс распознавания должен осуществляться надежно, невзирая на изменения освещенности и позы. Люди способны легко распознавать объекты, несмотря на то, что их внешний вид существенно изменяется, даже если судить по данным о значениях яркости пикселей на изображениях этих объектов. Например, мы всегда способны узнать лицо друга при разных условиях освещения или под разными углами зрения. В качестве еще более простого примера рассмотрим задачу распознавания рукописной цифры 6. Люди способны решить такую задачу независимо от изменения размеров и положения такого объекта на изображении, а также несмотря на небольшие изменения угла поворота³ надписи, изображающей эту цифру.

На данном этапе необходимо сделать одно важное замечание — геометрические трансформации, такие как перенос, масштабирование и поворот, или трансформации яркости изображения, вызванные физическим перемещением источников света, имеют иной характер по сравнению с изменениями внутри категории, например, такими различиями, которыми характеризуются лица разных людей. Очевидно, что единственным способом получения информации о различных типах человеческих лиц или о разных способах написания цифры 4 является обучение. С другой стороны, влияния геометрических и физических трансформаций носят систематический характер, поэтому должна существовать возможность исключить их из рассмотрения на основе продуманного проектирования состава характеристик, используемых для представления обучающих экземпляров.

Практика показала, что одним из весьма эффективных методов обеспечения инвариантности по отношению к геометрическим трансформациям является предварительная обработка рассматриваемого участка изображения и приведение его к стандартной позиции, масштабу и ориентации. Еще один вариант состоит в том, что можно просто игнорировать причинный характер геометрических и физических трансформаций, рассматривая их как дополнительные источники изменчивости изображений, поступающих в классификатор. В таком случае в обучающее множество необходимо включить экземпляры, соответствующие всем этим вариантам, в расчете на то, что классификатор выявит логическим путем данные о соответствующем множестве трансформаций входных данных, что позволит исключить из рассмотрения указанные причины изменения внешнего вида экземпляров.

Теперь перейдем к описанию конкретных алгоритмов распознавания объектов. Для упрощения сосредоточимся на задаче, постановка которой определена в двумерной системе координат, — и обучающие, и тестовые примеры заданы в форме двумерных растровых изображений. Очевидно, что данный подход вполне применим в таких областях, как распознавание рукописного текста. Но даже в случае трехмерных объектов может оказаться эффективным подход, предусматривающий использование способа представления этих объектов с помощью многочисленных двумерных изображений (рис. 24.18) и классификации новых объектов путем сравнения их с хранимыми изображениями (т.е. с некоторыми другими данными, представляющими те же объекты).

³ Ставить перед собой задачу добиться надежного распознавания при любых углах поворота не нужно и не желательно, поскольку цифру 6 можно повернуть так, что она станет похожей на цифру 9!



Рис. 24.18. Многочисленные изображения двух трехмерных объектов в разных видах

Как было описано в предыдущем разделе, для извлечения из изображения информации о трехмерных объектах в сцене могут использоваться многочисленные признаки. Кроме того, многочисленные признаки лежат в основе распознавания объектов, например, тигра можно узнать, заметив оранжевые и черные цвета на его шкуре, обнаружив на ней полосы или определив форму его тела.

Цвет и текстуру можно представить с использованием гистограмм или эмпирических распределений частот. Получив в качестве образца изображение тигра, можно определить, каково процентное соотношение количества пикселей, относящихся к разным цветам. В дальнейшем, после получения экземпляра с неизвестной классификацией, можно провести сравнение гистограммы его цветов с данными о полученных ранее примерах изображений тигра. Для анализа текстуры рассматриваются гистограммы, полученные в результате свертки изображения с фильтрами, имеющими различные ориентации и масштабы, после чего отыскиваются совпадения.

Как оказалось, задача использования формы для распознавания объектов является более сложной. Вообще говоря, существуют два основных подхода: **распознавание с учетом яркости**, в котором непосредственно используются значения яркости пикселей, и **распознавание с учетом характеристик**, в котором предусматривается применение данных о пространственном расположении извлеченных из изображения характеристик, таких как края или ключевые точки. После более подробного описания двух этих подходов мы рассмотрим также проблему **оценки позы**, т.е. проблему определения местонахождения и ориентации объектов в сцене.

Распознавание с учетом яркости

При таком подходе за основу берется подмножество пикселей изображения, которое соответствует распознаваемому объекту, и определяются данные о характеристиках как данные о самих исходных значениях яркости пикселей. Еще один вариант этого метода состоит в том, что вначале может быть выполнена свертка изображения с различными линейными фильтрами, после чего значения пикселей в результирующих

изображениях рассматриваются как характеристики. Как было показано в разделе 20.7, такой подход оказался очень успешным при решении таких задач, как распознавание рукописных цифр.

Для создания детекторов лиц, позволяющих распознавать лица с помощью баз данных с изображениями, использовался целый ряд статистических методов, включая методы на основе нейронных сетей с необработанными входными данными, представленными в виде характеристик пикселей; деревья решений с характеристиками, определяемыми с помощью различных фильтров полос и краев; а также наивные байесовские модели с характеристиками небольшого волнения (ряби). Некоторые результаты применения последнего метода показаны на рис. 24.19.

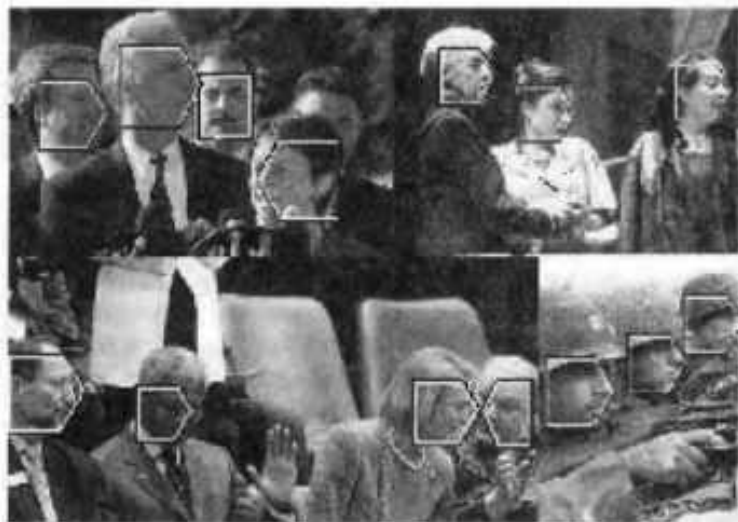


Рис. 24.19. Выходные данные алгоритма поиска лиц (любезно предоставлено Генри Шнейдерманом (Henry Schneiderman) и Такео Канаде (Takeo Kanade))

Одним из недостатков метода, в котором в качестве векторов характеристик используются необработанные данные о пикселах, является большая избыточность, свойственная этому способу представления. Предположим, что рассматриваются два пиксела, расположенные рядом на изображении щеки лица какого-то человека; между ними, скорее всего, будет весьма высокая корреляция, поскольку для них свойственны аналогичное геометрическое расположение, освещенность и т.д. Для сокращения количества размерностей вектора характеристик можно с успехом применять методы уменьшения объема данных, такие как анализ наиболее важных компонентов; использование таких методов обеспечивает распознавание объектов, подобных лицам, с большим быстродействием по сравнению с тем, которое может быть достигнуто с использованием пространства большей размерности.

Распознавание с учетом характеристик

Вместо применения в качестве характеристик необработанных данных о яркости пикселей можно использовать способы обнаружения и разметки пространственно

локализованных характеристик, таких как участки и края (см. раздел 24.3). Применение краев является целесообразным по двум описанным ниже важным причинам. Одной из них является уменьшение объема данных, связанное с тем, что количество краев намного меньше по сравнению с количеством пикселей изображения. Вторая причина обусловлена возможностью добиться инвариантности освещенности, поскольку края (при наличии подходящего диапазона контрастов) обнаруживаются приблизительно в одних и тех же местах, независимо от точной конфигурации освещенностей. Края представляют собой одномерные характеристики; были также предприняты попытки использовать двухмерные характеристики (участки) и нульмерные характеристики (точки). Обратите внимание на то, как отличаются трактовки пространственного расположения в подходах с учетом яркости и с учетом характеристик. В подходах с учетом яркости эти данные кодируются неявно, как индексы компонентов вектора характеристик, а в подходах с учетом характеристик характеристикой является само местонахождение (x, y) .

Неотъемлемым свойством любого объекта является инвариантное расположение краев; именно по этой причине люди могут легко интерпретировать контурные рисунки (см. рис. 24.13), даже несмотря на то, что подобные изображения не встречаются в природе! Простейший способ использования этих знаний основан на классификаторе по ближайшим соседним точкам. При этом предварительно вычисляются и сохраняются данные о конфигурациях краев, соответствующие представлениям всех известных объектов. А после получения конфигурации краев, соответствующей неизвестному объекту на изображении, являющимся предметом запроса, можно определить “расстояние” этого объекта от каждого элемента библиотеки хранимых представлений. После этого классификатор по ближайшим соседним точкам выбирает наиболее близкое соответствие.

Для описания понятия расстояния между изображениями было предложено много разных определений. Один из наиболее интересных подходов основан на идее **согласования с учетом деформации**. В своей классической работе *On Growth and Form* [1506] Дарси Томпсон заметил, что близкие, но не идентичные формы часто можно деформировать в подобные друг другу формы с использованием простых координатных преобразований⁴. При таком подходе понятие подобия формы воплощается на практике в виде следующего трехэтапного процесса: во-первых, ищется решение задачи соответствия между двумя формами, во-вторых, данные о соответствии используются для определения преобразования, позволяющего сделать эти формы аналогичными, и, в-третьих, вычисляется расстояние между двумя формами как сумма ошибок согласования между соответствующими точками, наряду с термом, в котором измеряется величина выравнивающего преобразования.

Форма представляется с помощью конечного множества точек, полученных в виде выборки, взятой на внутренних или внешних контурах формы. Эти данные могут быть получены как сведения о местонахождениях пикселей краев, обнаруженные детектором краев, и представлены в виде множества $\{p_1, \dots, p_N\}$ из N точек. Примеры множеств точек, соответствующих двум формам, приведены на рис. 24.20, а, б.

⁴ В современной компьютерной графике такой процесс называется **трансформацией**.

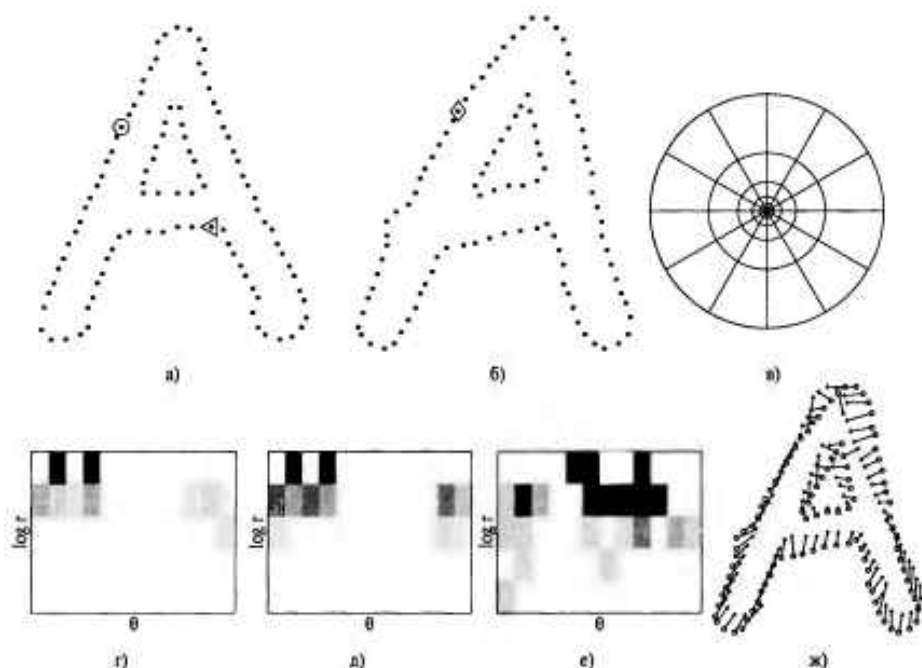


Рис. 24.20. Процедура вычисления и согласования контекстов формы: множества точек краев двух форм (а), (б); схема секторов логарифмической–полярной гистограммы, используемой при вычислении контекстов формы. Используется 5 секторов для $\log r$ и 12 секторов для θ (в); примеры контекстов формы для опорных образцов, отмеченных точками со значками \circ , ϕ , \triangleleft на рис. 24.20, а, б. Каждый контекст формы представляет собой логарифмическую–полярную гистограмму координат остальных точек множества, измеренных с использованием опорной точки в качестве начала координат (затемненные ячейки означают, что в данном секторе имеется больше одной точки). Обратите внимание на внешнее подобие контекстов формы знакам \circ и ϕ , поскольку эти контексты были вычислены для относительно подобных точек в двух формах. В отличие от этого, контекст формы для знака \triangleleft существенно отличается (г)–(е); соответствия между рис. 24.20, а, б, обнаруженные путем согласования двухдольных графов, с указанием стоимостей преобразования, определяемых на основе расстояния χ^2 между гистограммами (ж)

Теперь рассмотрим конкретную точку выборки p_i , наряду с множеством всех векторов, исходящих из этой точки в направлении всех других точек выборки в форме. Эти векторы представляют конфигурацию всей формы относительно рассматриваемой опорной точки. Такое представление лежит в основе следующей идеи: с каждой точкой выборки можно связать дескриптор, или **контекст формы**, который приближенно представляет расположение остальной части формы по отношению к данной точке. Точнее, контекст формы точки p_i представляет собой приближенную пространственную гистограмму h_i относительных координат $p_k - p_i$ остальных $N-1$ точек p_k . Для определения сегментов используется логарифмическая–полярная система координат, обеспечивающая то, что дескриптор становится более чувствительным к различиям в ближайших друг к другу пикселах. Пример расположения сегментов показан на рис. 24.20, в.

Обратите внимание на то, что неотъемлемым свойством этого определения контекста формы является его инвариантность к операции переноса, поскольку все изменения выполняются по отношению к точкам в объекте. Для достижения инвари-

антности к операции масштабирования все радиальные расстояния нормализуются путем деления на среднее расстояние между парами точек.

Контексты формы позволяют решить задачу установления соответствия между двумя аналогичными, но не идентичными формами, наподобие тех, которые показаны на рис. 24.20, *а, б*. Контексты формы являются разными для различных точек на одной и той же форме S , тогда как соответствующие (гомологичные) точки на подобных формах S и S' , как правило, имеют одинаковые контексты формы. Таким образом, задача поиска соответствующих друг другу точек двух форм преобразована в задачу поиска партнеров, имеющих взаимно подобные контексты формы.

Точнее, рассмотрим точку p_i на первой форме и точку q_j на второй форме. Допустим, что $C_{ij} = C(p_i, q_j)$ обозначает стоимость согласования этих двух точек. Поскольку контексты формы представляют собой распределения, выраженные в виде гистограмм, вполне обоснован подход, предусматривающий использование расстояния χ^2 , следующим образом:

$$C_{ij} = \frac{1}{2} \sum_{k=1}^K \frac{[h_i(k) - h_j(k)]^2}{h_i(k) + h_j(k)}$$

где $h_i(k)$ и $h_j(k)$ обозначают k -й сектор нормализованных гистограмм в точках p_i и q_j . Если дано множество стоимостей C_{ij} согласования между всеми парами точек i на первой форме и точек j на второй форме, то может быть принято решение минимизировать общую стоимость согласования с учетом ограничения, что это согласование должно выполняться на основе взаимно-однозначного соответствия. Это — пример задачи поиска паросочетаний взвешенного двухдольного графа, которая может быть решена за время $O(N^3)$ с использованием так называемого венгерского алгоритма (Hungarian algorithm).

Если известны соответствия в точках выборки, то данные о соответствии можно распространить на всю форму, оценивая стоимость согласующего преобразования, которое позволяет отобразить одну форму на другой. Особенно эффективным является подход с использованием регуляризованного тонкостенного сплайна (regularized thin plate spline). После того как формы будут согласованы, задача вычисления оценок подобия становится относительно несложной. Расстояние между двумя формами может быть определено как взвешенная сумма расстояний контекстов форм между соответствующими точками и как энергия изгиба, связанная с тонкостенным сплайном. После получения такой меры расстояния для решения задачи распознавания можно использовать простой классификатор по ближайшим соседним точкам. Превосходная иллюстрация, демонстрирующая высокую эффективность применения этого подхода при классификации рукописных цифр, приведена в главе 20.

Оценка позы

Интерес представляет не только задача определения того, каковым является некоторый объект, но и задача определения его позы, т.е. его позиции и ориентации по отношению к наблюдателю. Например, при решении проблемы манипулирования объектами на производстве приходится учитывать, что захват робота не может взять объект до тех, пока не будет известна его поза. В случае твердых объектов, как

трехмерных, так и двухмерных, эта проблема имеет простое и полностью определенное решение, основанное на ~~на~~ **методе выравнивания**, который описан ниже.

В этом методе объект представляется с помощью M характеристик, или различных точек m_1, m_2, \dots, m_M в трехмерном пространстве (в качестве таковых могут, допустим, рассматриваться вершины многогранного объекта). Координаты этих точек измеряются в некоторой системе координат, наиболее подходящей для данного объекта. После этого точки подвергаются операции трехмерного вращения \mathbf{R} с неизвестными параметрами, за которой следует операция переноса на неизвестную величину \mathbf{t} , а затем выполняется операция проекции, которая приводит к появлению точек характеристик изображения p_1, p_2, \dots, p_N на плоскости изображения. Вообще говоря, $N \neq M$, поскольку некоторые точки модели могут закрывать друг друга, а детектор характеристик может пропустить некоторые характеристики (или выявить ложные характеристики, появление которых обусловлено наличием шума). Такое преобразование для трехмерной модели точек m_i и соответствующих точек изображения p_i можно представить следующим образом:

$$p_i = \Pi (\mathbf{R}m_i + \mathbf{t}) = Q(m_i)$$

где \mathbf{R} — матрица вращения, \mathbf{t} — вектор переноса; Π — перспективная проекция или одно из ее приближений, такое как масштабируемая ортогональная проекция. Чистым результатом становится трансформация Q , которая приводит точки модели m_i в соответствие с точками изображения p_i . При этом, хотя первоначально трансформация Q не определена, известно, что (применительно к твердотельным объектам) Q должна быть одинаковой для всех точек модели.

Задачу определения преобразования Q можно решить, получив значения трехмерных координат трех точек модели и их двухмерных проекций. В основе этого подхода лежит следующая интуитивная идея: могут быть легко составлены уравнения, связывающие координаты p_i с координатами m_i . В этих уравнениях неизвестные величины соответствуют матрице вращения \mathbf{R} и вектору переноса \mathbf{t} . Если количество уравнений достаточно велико, то возможность получения решения для Q становится неоспоримой. Мы не будем приводить здесь доказательство этой гипотезы, а просто сформулируем следующий результат.

Если даны три точки, m_1, m_2 и m_3 , в модели, не лежащие на одной прямой, и их масштабированные ортогональные проекции, p_1, p_2 и p_3 , на плоскости изображения, то существуют две и только две трансформации из системы координат трехмерной модели в систему координат двухмерного изображения.

Эти трансформации связаны друг с другом, поскольку зеркально противоположны относительно плоскости изображения; они могут быть вычислены на основе простого решения в замкнутой форме. Если существует возможность идентифицировать характеристики модели, соответствующие трем характеристикам в изображении, то может быть вычислена Q — поза объекта. В предыдущем подразделе обсуждался метод определения соответствий с использованием согласования контекстов формы. Если же объект имеет четко определенные углы или другие заметные точки, то становится доступным еще более простой метод. Идея его состоит в том, что необходимо повторно формировать и проверять соответствия. Мы должны выдвинуть первоначальную гипотезу о соответствии тройки точек изображения тройке точек модели и использовать функцию Find-Transform для формирования гипотезы Q .

Если принятое предположение о соответствии было правильным, то трансформация Q является правильной и после ее применения к оставшимся точкам модели приводит к получению предсказания координат точек изображения. Если принятое предположение было неправильным, то трансформация Q также является неправильной и после ее применения к оставшимся точкам модели не позволяет предсказывать координаты точек изображения.

Описанный выше подход лежит в основе алгоритма *Align*, приведенного в листинге 24.1. Этот алгоритм находит позу для данной конкретной модели или возвращает индикатор неудачи. Временная сложность данного алгоритма в худшем случае пропорциональна количеству сочетаний троек точек модели и троек точек изображения, или

$$\binom{N}{3} \binom{M}{3},$$

умноженному на стоимость проверки каждого сочетания. Стоимость проверки пропорциональна $M \log N$, поскольку необходимо предсказывать позицию изображения для каждой из M точек модели и находить расстояние до ближайшей точки изображения, что требует выполнения $\log N$ операций, если точки изображения представлены с помощью некоторой подходящей структуры данных. Поэтому в наихудшем случае сложность алгоритма выравнивания определяется значением $O(M^4 N^3 \log N)$, где M и N — количество точек модели и изображения соответственно. Методы, основанные на принципе кластеризации поз в сочетании со средствами рандомизации, позволяют уменьшить сложность до $O(MN^3)$. Результаты применения этого алгоритма к изображению степлера показаны на рис. 24.21.

Листинг 24.1. Неформальное описание алгоритма выравнивания

```

function Align(image, model) returns решение solution или индикатор
    неудачи failure
    inputs: image, список координат характеристик изображения
             model, список координат характеристик модели

    for each ( $p_1, p_2, p_3$ ) in Triplets(image) do
        for each ( $m_1, m_2, m_3$ ) in Triplets(model) do
             $Q \leftarrow \text{Find-Transform}((p_1, p_2, p_3), (m_1, m_2, m_3))$ 
            if проекция, соответствующая гипотезе  $Q$ ,
                позволяет распознать изображение then
                return  $Q$ 

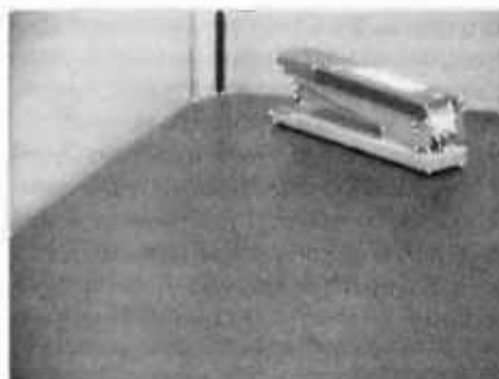
    return failure

```

24.6. ИСПОЛЬЗОВАНИЕ СИСТЕМЫ МАШИННОГО ЗРЕНИЯ ДЛЯ МАНИПУЛИРОВАНИЯ И ПЕРЕДВИЖЕНИЯ

Одно из наиболее важных направлений использования систем машинного зрения состоит в получении информации как для манипулирования объектами (определения их местоположения, захвата, изменения их положения в пространстве и т.д.), так и для передвижения без столкновений с препятствиями. Способность использовать зрение для этих целей присуща системам зрения даже самых примитив-

ных животных. Во многих случаях по своему устройству такая система зрения состоит из минимально необходимого набора компонентов; под этим подразумевается, что она извлекает из доступного светового поля только такую информацию, которая требуется животному для организации своего поведения. Вполне возможно, что системы зрения наиболее высокоразвитых животных стали результатом эволюции, которая началась с появления на одном конце тела у самых ранних, примитивных организмов светочувствительного пятна, с помощью которого они устремлялись к свету (или прятались от него). Как было описано в разделе 24.4, в нервной системе мухи существует очень простая система распознавания оптического потока, позволяющая мухе садиться на стены. В классическом исследовании *What the Frog's Eye Tells the Frog's Brain* [914] сделано следующее замечание в отношении лягушки: "Она умерла бы с голоду, окруженная пищей, если бы эта пища не двигалась. Лягушка выбирает пищу только после определения ее размеров и движения".



а)



б)

Рис. 24.21. Результаты применения алгоритма выравнивания к изображению степлера; углы, найденные на изображении степлера (а); гипотетическая реконструкция, наложенная на первоначальное изображение (любезно предоставлено Кларком Олсоном (Clark Olson)) (б)

Системы машинного зрения используются в "организмах", называемых роботами. Рассмотрим особую разновидность робота — автоматизированное транспортное средство, движущееся по шоссе (рис. 24.22). Вначале проанализируем стоящие перед нами задачи, затем определим, какие алгоритмы машинного зрения позволят нам получить информацию, необходимую для успешного выполнения этих задач. Ниже перечислены задачи, с которыми сталкивается водитель.

1. Управление движением в поперечном направлении. Обеспечение того, чтобы транспортное средство надежно придерживалось своей полосы движения или плавно переходило на другую полосу движения в случае необходимости.
2. Управление движением в продольном направлении. Обеспечение того, чтобы постоянно соблюдалась безопасная дистанция до транспортного средства, идущего впереди.
3. Предотвращение столкновений с препятствиями. Слежение за транспортными средствами на соседних полосах движения и подготовка к маневрам, необходимым для предотвращения столкновения, если водитель одного из них решит перейти на другую полосу движения.



Рис. 24.22. Изображение дороги, снятое видеокамерой, которая расположена внутри автомобиля. Горизонтальными белыми полосками обозначены окна поиска, в пределах которых контроллер отыскивает маркировку полос движения. Низкое качество изображения является вполне типичным для черно-белого видеосигнала с низким разрешением

Перед водителем стоит проблема — определить и осуществить подходящие действия по изменению направления движения, ускорению и торможению, позволяющие наилучшим образом выполнить стоящие перед ним задачи.

Что касается управления движением в поперечном направлении, то для этого необходимо постоянно обновлять данные о положении и ориентации автомобиля относительно его полосы движения. Применительно к изображению, показанному на рис. 24.22, для поиска краев, соответствующих сегментам маркировки полосы движения, можно использовать алгоритмы распознавания краев. После этого с данными элементами представления краев можно согласовать гладкие кривые. Параметры этих кривых несут информацию о поперечном положении автомобиля, направлении, в котором он движется относительно своей полосы движения, и о кривизне самой полосы движения. Эта информация, наряду с информацией о динамике автомобиля, включает в себе все необходимое для системы рулевого управления. Следует также отметить, что от одного видеокadra к другому происходит лишь небольшое изменение в положении проекции полосы движения на изображении, поэтому уже известно, где искать на изображении маркировку полосы движения; например, на данном рисунке достаточно рассмотреть только те участки, которые обозначены параллельными белыми полосками.

А что касается управления движением в продольном направлении, то необходимо знать расстояния до идущих впереди транспортных средств. Для получения такой информации могут использоваться бинокулярные стереоданные или оптический поток. Оба эти подхода могут быть упрощены с использованием ограничений проблемной области, определяемых тем фактом, что вождение происходит на плоской поверхности. В настоящее время автомобили, действующие под управлением систем

машинного зрения, в которых используются эти методы, показали свою способность двигаться в течение продолжительных периодов времени на максимальных скоростях, разрешенных на автомагистралях.

Приведенный выше пример решения проблемы вождения позволяет очень четко подчеркнуть одну мысль: *«Для решения конкретной задачи нет необходимости извлекать из изображения всю информацию, которая может быть в принципе получена с его помощью»*. Не требуется восстанавливать точную форму каждого встречного или попутного автомобиля, решать задачу определения формы на основании текстуры для поверхности травы, растущей вдоль автомагистрали, и т.д. Потребности данной задачи определяют необходимость в получении лишь информации определенных видов, и поэтому можно добиться значительного повышения скорости вычислений и надежности, восстанавливая только эту информацию и в полной мере применяя ограничения проблемной области. Наша цель при обсуждении общих подходов, представленных в предыдущем разделе, состояла в демонстрации того, что они формируют общую теорию, которую можно специализировать в интересах решения конкретных задач.

24.7. РЕЗЮМЕ

Хотя на первый взгляд кажется, что люди осуществляют действия по восприятию без каких-либо усилий, для обеспечения восприятия требуется большой объем сложных вычислений. Задача зрения состоит в извлечении информации, необходимой для решения таких задач, как манипулирование, навигация и распознавание объектов.

- Геометрические и физические аспекты процесса **формирования изображения** глубоко изучены. Если дано описание трехмерной сцены, можно легко сформировать ее изображение из любой произвольной позиции видеокамеры (это — задача компьютерной графики). Задача организации обратного процесса, в котором происходит переход от изображения к описанию сцены, является более сложной.
- Для извлечения визуальной информации, необходимой для решения задач манипулирования, навигации и распознавания, необходимо создавать промежуточные представления. В ранних алгоритмах **обработки изображения** для систем машинного зрения предусматривалось извлечение из изображения таких примитивных характеристик, как края и участки.
- В каждом изображении имеется целый ряд признаков, позволяющих получить информацию о конфигурации рассматриваемой трехмерной сцены: движение, стереоданные, текстура, затенение и контуры. Выделение каждого из этих признаков основано на исходных допущениях о физических сценах, позволяющих добиваться почти полностью непротиворечивых интерпретаций.
- Задача распознавания объектов в своей полной постановке является весьма сложной. В данной главе рассматривались подходы к решению этой задачи с учетом яркости и характеристик. Кроме того, в настоящей главе приведен простой алгоритм оценки позы. Существуют и другие возможности.

БИБЛИОГРАФИЧЕСКИЕ И ИСТОРИЧЕСКИЕ ЗАМЕТКИ

Упорные попытки понять, как функционирует зрение человека, предпринимались с самых древних времен. Евклид (около 300 г. до н.э.) в своих трудах писал о естественной перспективе — об отображении, которое связывает с каждой точкой P в трехмерном мире направление луча OP , соединяющего центр проекции O с точкой P . Он также был хорошо знаком с понятием параллакса движения. Следующий значительный этап развития математической трактовки перспективной проекции, на этот раз в контексте проекции на плоские поверхности, наступил в XV веке в Италии, в период Возрождения. Создателем первых рисунков, основанных на геометрически правильной проекции трехмерной сцены, принято считать Брунеллески (1413 год). В 1435 году Альберти составил свод правил построения перспективной проекции, ставший источником вдохновения для многих поколений художников, чьи художественные достижения восхищают нас и поныне. Особенно весомый вклад в развитие науки о перспективе (как она называлась в те времена) внесли Леонардо да Винчи и Альбрехт Дюрер. Составленные Леонардо в конце XV столетия описания игры света и тени (светотени), теневых и полутеневых областей затенения, а также воздушной перспективы до сих пор не потеряли своего значения [790].

Хотя знаниями о перспективе владели еще древние греки, в их воззрениях присутствовала забавная путаница, поскольку они неправильно понимали, какую роль играют глаза в процессе зрения. Аристотель считал, что глаза — это устройства, испускающие лучи, что соответствует современным представлениям о работе лазерных дальномеров. Этим ошибочным взглядом положили конец труды арабских ученых X столетия, в частности Альхазена. В дальнейшем началась разработка камер-обскура различных видов. На первых порах они представляли собой комнаты (камера-обскура по латыни — “темная комната”), в которые свет попадал через малое отверстие в одной стене, а на противоположной стене создавалось изображение сцены, происходящей наружи. Безусловно, во всех этих камерах изображение было перевернутым, что вызывало невероятное смущение современников. Ведь если глаз рассматривать как аналогичный такому устройству формирования изображения, как камера-обскура, то почему же мы видим предметы такими, каковы они на самом деле? Эта загадка не давала покоя величайшим умам той эпохи (включая Леонардо). Окончательно решить эту проблему удалось лишь благодаря работам Кеплера и Декарта. Декарт поместил препарат глаза, с задней стенки которого была удалена непрозрачная оболочка, в отверстие оконного ставня. В результате было получено перевернутое изображение, сформировавшееся на куске бумаги, заменившем сетчатку. Хотя изображение на сетчатке глаза действительно перевернуто, такая ситуация не вызывает проблемы, поскольку мозг интерпретирует полученное изображение правильно. Говоря современным языком, для этого достаточно обеспечить правильный доступ к структуре данных.

Очередные крупные успехи в изучении зрения были достигнуты в XIX веке. Благодаря трудам Гельмгольца и Вундта, описанным в главе 1, методика проведения психофизических экспериментов стала строгой научной дисциплиной. А труды Юнга, Максвелла и Гельмгольца привели к созданию трехкомпонентной теории цветоощущения. Стереоскоп, изобретенный Витстоуном [1582], позволил продемонстрировать, что люди получают возможность определять глубину изображения,

если на левый и правый глаз поступают немного разные картинки. После того как стало известно о создании стереоскопа, этот прибор быстро завоевал популярность в гостиных и салонах по всей Европе. Возникла новая научная область — *фотограмметрия*, основанная на принципиально важном понятии бинокулярных стереоданных, согласно которому два изображения сцены, снятые немного с разных точек зрения, несут достаточную информацию для получения трехмерной реконструкции сцены. В дальнейшем были получены важные математические результаты; например, Круппа [861] доказал, что если даны два изображения пяти различных точек одного и того же объекта, то можно реконструировать данные о повороте и переносе камеры с одной позиции в другую, а также о глубине сцены (с точностью до коэффициента масштабирования). Хотя геометрия стереоскопического зрения была известна уже давно, не было ясно, как решают задачу фотограмметрии люди, автоматически согласующие соответствующие точки изображений. Удивительные способности людей решать проблему соответствия были продемонстрированы Юлешем [755], который изобрел стереограмму, состоящую из случайно выбранных точек. На решение проблемы соответствия как в машинном зрении, так и в фотограмметрии в 1970-х и в 1980-х годах были потрачены значительные усилия.

Вторая половина XIX столетия была основным периодом становления области психофизических исследований человеческого зрения. В первой половине XX столетия наиболее значительные результаты исследований в области зрения были получены представителями школы гештальт-психологии, возглавляемой Максом Вертхеймером. Эти ученые были проводниками взглядов, что основными единицами восприятия должны быть законченные формы, а не их компоненты (такие как края), и выдвинули лозунг: “Целое не равно сумме его частей”.

Период исследований после Второй мировой войны характеризуется новым всплеском активности. Наиболее значительной была работа Дж.Дж. Гибсона [551], [552], который подчеркнул важность понятий оптического потока, а также градиентов текстуры в оценке таких переменных описания внешней среды, как поворот и наклон поверхности. Гибсон еще раз подчеркнул значимость стимулов и их разнообразия. Например, в [553] указано, что поле оптического потока всегда содержит достаточно информации для определения самодвижения наблюдателя по отношению к его среде. В сообществе специалистов по системам компьютерного зрения основные работы в этой области и в (математически эквивалентной) области выявления структуры по данным о движении проводились главным образом в 1980-х и в 1990-х годах. Наиболее яркими проявлениями этой деятельности стали оригинальные работы [815], [945] и [1526]. Возникавшая на первых порах озабоченность в отношении стабильности структуры, выявленной на основании данных о движении, была полностью развеяна благодаря работе Томази и Канаде [1511], которые показали, что форма может быть восстановлена абсолютно точно благодаря использованию многочисленных кадров и получаемой в результате этого широкой базисной линии.

В [230] описано удивительное устройство системы зрения мухи и показано, что это насекомое обладает остротой временного визуального восприятия, в десять раз лучшей по сравнению с человеком. Это означает, что муха способна смотреть фильм, воспроизводимый с частотой до 300 кадров в секунду, различая при этом отдельные кадры.

Принципиально важным нововведением, представленным в исследованиях, которые проводились в 1990-х годах, было выявление с помощью обучения проектив-

ной структуры по данным о движении. Как показано в [452], при таком подходе не требуется калибровка видеокамеры. Это открытие тесно связано с работами, послужившими основой для использования геометрических инвариантов при распознавании объектов, обзор которых приведен в [1104], и с работами по разработке аффинной структуры по данным о движении [816]. В 1990-х годах анализ движения нашел много новых областей применения благодаря значительному увеличению быстроедействия и объема памяти компьютеров, а также широкому распространению цифровой видеоаппаратуры. Особенно важное применение нашли методы создания геометрических моделей сцен реального мира, которые предназначены для формирования изображений с помощью средств компьютерной графики; эти работы привели к созданию алгоритмов реконструкции наподобие тех, которые представлены в [364]. В [454] и [626] приведено исчерпывающее описание геометрии множественных представлений.

В области компьютеризированных систем зрения наиболее важными основополагающими работами по логическому выводу формы на основании текстуры были [60] и [1461]. Они были посвящены описанию плоских поверхностей, а для криволинейных поверхностей результаты исчерпывающего анализа приведены в [518] и [973].

В сообществе специалистов в области компьютерных систем зрения проблема логического вывода формы из данных о затенении была впервые исследована Бертольдсом Хорном [676]. В [678] представлен исчерпывающий обзор основных статей в этой области. В указанном научном направлении было принято принимать целый ряд упрощающих допущений, из которых наиболее важным было игнорирование влияния взаимного освещения. Важность проблемы взаимного освещения была впервые осознана в сообществе специалистов по компьютерной графике, которые стремились точно разрабатывать модели трассировки лучей и диффузного отражения, чтобы учесть наличие взаимного освещения. С критикой основных теоретических и эмпирических подходов в этой области можно ознакомиться в [484].

В области логического вывода информации о форме по данным о контурах самый первый, решающий вклад был сделан Хаффменом [702] и Клоувсом [271], после чего Маккуорт [966] и Сугихара [1473] провели до конца анализ методов, применимых к многогранным объектам. Малик [971] разработал схему разметки для кусочно-гладких криволинейных объектов. В [801] показано, что задача разметки линий для трехгранных сцен является NP-полной.

Для правильной трактовки визуальных эффектов, возникающих в проекциях гладких криволинейных объектов, требуется совместное использование дифференциальной геометрии и теории особенностей. Наилучшим исследованием на эту тему является книга Кендеринка *Solid Shape* [814].

Оригинальной работой по распознаванию трехмерных объектов явились тезисы Робертса [1294], опубликованные в Массачусеттском технологическом институте (Massachusetts Institute of Technology — MIT). Эту работу многие считают первыми тезисами докторской диссертации по машинному зрению; в ней впервые представлено несколько ключевых идей, в том числе касающихся обнаружения краев и согласования на основе моделей. Метод обнаружения краев Кэнни был представлен в [218]. Идея выравнивания, также впервые выдвинутая Робертсом, снова вышла на передний план в 1980-х годах после опубликования работ [711] и [950]. Значительное повышение эффективности методов оценки позы путем выравнивания было достигнуто Олсоном [1156]. Еще одним важным направлением исследований в области распознавания

трехмерных объектов явился подход, основанный на идее описания форм в терминах объемных примитивов на основе **обобщенных цилиндров**, который был предложен Томом Бинфордом [128] и нашел особенно широкое распространение.

Исследования в области машинного зрения, посвященные распознаванию объектов, в основном сосредоточивались на проблемах, возникающих в результате получения проекции трехмерных объектов в виде двухмерных изображений, а в сообществе специалистов по распознаванию образов существовала другая традиция, в которой эта задача рассматривалась как относящаяся к области классификации образов. Этих специалистов в основном интересовали примеры, относящиеся к таким проблемным областям, как оптическое распознавание символов и распознавание рукописных почтовых кодов, в которых основные усилия были направлены на изучение характеристик типичных вариаций искомого класса объектов и отделение этих объектов от объектов других классов. Сравнение таких подходов приведено в [904]. К другим работам по распознаванию объектов относятся исследования по распознаванию лиц [1422] и [1543]. В [98] описан подход на основе контекста формы. В [395] впервые показаны результаты разработки автомобиля с визуальным управлением для автоматического вождения по автомагистралям на высоких скоростях; в [1224] показаны результаты достижения аналогичной производительности с использованием подхода на основе нейронной сети.

Наилучшее и самое полное описание человеческого зрения можно найти в книге Стивена Палмера *Vision Science: Photons to Phenomenology* [1167]; а книги Дэвида Хабела *Eye, Brain and Vision* [700] и Ирвина Рока *Perception* [1300] представляют собой краткие введения, в основном посвященные соответственно нейрофизиологии и восприятию.

В настоящее время наиболее всесторонним учебником для специалистов по машинному зрению является книга Дэвида Форсита (David Forsyth) и Джин Понсе (Jean Ponce) *Computer Vision: A Modern Approach*. Значительно более краткие описания можно найти в [1111] и [1513]. Интерес представляют также два изданных немного раньше, но все еще значимых учебника, в каждом из которых рассматривается ряд специальных тем: *Robot Vision* [677] и *Three-Dimensional Computer Vision* [453]. Важную роль в объединении усилий специалистов по машинному зрению и специалистов по более традиционным областям биологического зрения (психофизике и нейробиологии) в свое время сыграла книга Дэвида Матта *Vision* [986]. Двумя основными журналами по машинному зрению являются *IEEE Transactions on Pattern Analysis and Machine Intelligence* и *International Journal of Computer Vision*. К числу конференций по машинному зрению относятся *ICCV* (International Conference on Computer Vision), *CVPR* (Computer Vision and Pattern Recognition) и *ECCV* (European Conference on Computer Vision).

УПРАЖНЕНИЯ

- 24.1. В тени дерева с плотной, густой кроной можно обнаружить множество пятен света. На удивление, все эти пятна кажутся круглыми. С чем это связано? Ведь в конечном итоге просветы между листьями, через которые проникают лучи солнца, вряд ли имеют круглую форму.

- 24.2.** Нанесите разметку на контурный рисунок, приведенный на рис. 24.23, приняв предположение, что все наружные края размечены как закрывающие и что все вершины являются трехгранными. Выполните эту задачу с помощью алгоритма перебора с возвратами, который проверяет вершины в последовательности A , B , C и D , выбирая на каждом этапе вариант, совместимый с размеченными ранее соединениями и краями. После этого попробуйте применить последовательность вершин B , D , A и C .

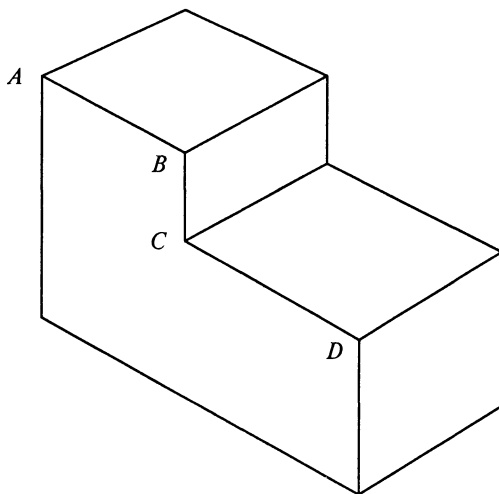


Рис. 24.23. Рисунок, подлежащий разметке, в котором все вершины являются трехгранными

- 24.3.** Рассмотрим цилиндр бесконечной длины с радиусом r , ось которого направлена вдоль оси y . Цилиндр имеет ламбертову поверхность и рассматривается с помощью видеокамеры, направленной вдоль положительной оси z . Что вы можете рассчитывать увидеть на этом изображении, если цилиндр освещается точечным источником света, находящимся на бесконечно большом расстоянии со стороны положительной полуоси x ? Объясните ваш ответ, нарисовав контуры равной яркости на спроектированном изображении. Являются ли контуры равной яркости расположенными через равномерные интервалы?
- 24.4.** Края на изображении могут соответствовать самым разнообразным визуальным эффектам, возникающим в сцене. Рассмотрите обложку данной книги и примите предположение, что это — картина реальной трехмерной сцены. Определите на этом изображении десять краев с различной яркостью и для каждого из них укажите, соответствует ли оно сосредоточенной неоднородности:
- а) по глубине;
 - б) по нормали к поверхности;
 - в) по отражательной способности;
 - г) по освещенности.

- 24.5. Покажите, что операция свертки с заданной функцией f является коммутативной по отношению к операции дифференцирования; иными словами, покажите, что $(f * g)' = f * (g)'$.
- 24.6. Рассматривается вопрос о возможности применения некоторой стереоскопической системы для составления карты местности. Она состоит из двух видеокамер CCD, в каждой из которых имеется 512×512 пикселей на квадратном датчике площадью 10×10 см. Применяемые линзы имеют фокусное расстояние 16 см, где фокус зафиксирован в бесконечности. Для соответствующих точек с координатами (u_1, v_1) на левом изображении и (u_2, v_2) на правом изображении верно, что $v_1 = v_2$, поскольку оси x двух плоскостей изображения параллельны эпиполярным линиям. Оптические оси этих двух видеокамер являются параллельными. Базисная линия между камерами равна 1 м.
- а) Если наименьшая дальность, которая должна быть измерена, равняется 16 м, то каково наибольшее рассогласование (в пикселах), которое может при этом возникнуть?
 - б) Какова разрешающая способность по дальности на расстоянии 16 м, которая обусловлена наличием интервала между пикселями?
 - в) Какая дальность соответствует рассогласованию в один пиксел?
- 24.7. Предположим, что нужно применить алгоритм выравнивания в промышленной установке, в которой по ремню конвейера движутся плоские детали и фотографируются видеокамерой, находящейся вертикально над ремнем конвейера. Позиция детали задается тремя переменными: одна из них определяет поворот, а две другие — положение относительно двух горизонтальных осей. Тем самым задача упрощается, а для функции Find-Transform требуется, чтобы позу определяли только две пары соответствующих характеристик изображения и модели. Определите сложность этой процедуры выравнивания в наилучшем случае.
- 24.8. (*Любезно предоставлено Пьетро Пероной (Pietro Perona).*) На рис. 24.24 показаны две видеокамеры в точках X и Y , с помощью которых ведется наблюдение за сценой. Нарисуйте изображение, поступающее на каждую видеокамеру, приняв предположение, что все обозначенные точки находятся на одной и той же горизонтальной плоскости. Можно ли с помощью этих двух изображений сделать заключение об относительных расстояниях точек A , B , C , D и E от базисной линии видеокамер? На чем должно быть основано это заключение?
- 24.9. Какие из приведенных ниже утверждений являются истинными и какие ложными?
- а) Обнаружение соответствующих друг другу точек в стереоскопических изображениях — самая простая стадия процесса стереоскопического поиска глубины.
 - б) Извлечение формы из текстуры можно выполнить, проектируя на сцену сетку световых полос.
 - в) Схема разметки Хаффмена–Клоувса предназначена для использования с любыми многогранными объектами.

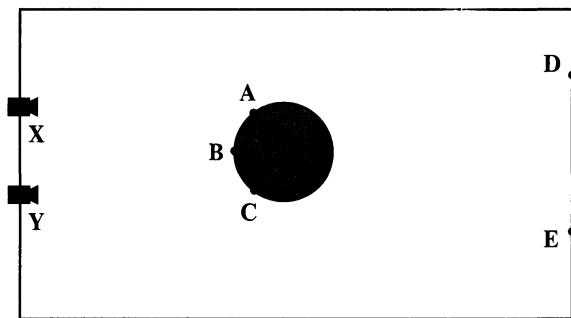


Рис. 24.24. Вид сверху системы машинного зрения с двумя видеокамерами, в которой ведется наблюдение за бутылкой и стоящей сзади ее стеной

- г) На контурных рисунках криволинейных объектов метка линии по мере прохождения от одного конца линии к другому может изменяться.
- д) При использовании стереоскопических изображений одной и той же сцены большая точность вычисления глубины достигается, если две камеры расположены дальше друг от друга.
- е) Проекциями линий равной длины в сцене всегда становятся линии равной длины в изображении.
- ж) Прямые линии в изображении обязательно соответствуют прямым линиям в сцене.

24.10. Фрагмент видеоизображения, приведенный на рис. 24.22, снят из автомобиля, находящегося на полосе движения, которая предназначена для выезда с автострасы. На полосе движения, расположенной непосредственно слева, видны два автомобиля. На каком основании наблюдатель мог бы заключить, что один из них ближе к нему, чем другой?

25 РОБОТОТЕХНИКА

В этой главе описано, как оснастить агентов физическими исполнительными механизмами, чтобы они могли немного пошалить.

25.1. ВВЕДЕНИЕ

☞ **Роботы** — это физические агенты, которые выполняют поставленные перед ними задачи, проводя манипуляции в физическом мире. Для этой цели роботов оснащают ☞ **исполнительными механизмами**, такими как ноги, колеса, шарниры и захваты. Исполнительные механизмы имеют единственное назначение — прилагать физические усилия к среде¹. Кроме того, роботов оснащают ☞ **датчиками**, которые позволяют им воспринимать данные об окружающей их среде. В современных роботах применяются различные виды датчиков, включая те, что предназначены для измерения характеристик среды (например, видеокамеры и ультразвуковые дальнометры), и те, которые измеряют характеристики движения самого робота (например, гироскопы и акселерометры).

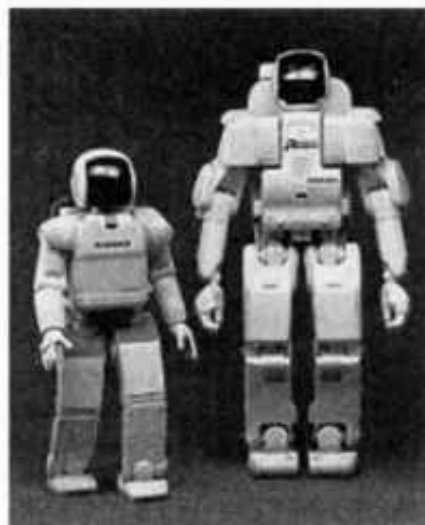
Большинство современных роботов относится к одной из трех основных категорий. ☞ **Роботы-манипуляторы**, или роботы-руки, физически привязаны к своему рабочему месту, например на заводском сборочном конвейере или на борту Международной космической станции. В движении манипулятора обычно участвует вся цепочка управляемых шарниров, что позволяет таким роботам устанавливать свои исполнительные механизмы в любую позицию в пределах своего рабочего пространства. Манипуляторы относятся к типу наиболее распространенных промышленных роботов, поскольку во всем мире установлено свыше миллиона таких устройств. Некоторые мобильные манипуляторы используются в больницах в качестве ассистентов хирургов. Без робототехнических манипуляторов в наши дни не смогут продолжать свою производственную деятельность большинство автомобильных заводов, а некоторые манипуляторы использовались даже для создания оригинальных художественных произведений.

¹ В главе 2 речь шла, скорее, об исполнительных системах (actuator), а не об исполнительных механизмах (effector). Исполнительной системой называется система управления, которая передает команды исполнительному механизму, а исполнительным механизмом называется само физическое устройство.

Ко второй категории относятся **мобильные роботы**. Роботы такого типа передвигаются в пределах своей среды с использованием колес, ног или аналогичных механизмов. Они нашли свое применение при доставке обедов в больницах, при перемещении контейнеров в грузовых доках, а также при выполнении аналогичных задач. В этой книге уже встречался один пример мобильного робота — **автоматическое наземное транспортное средство** (Unmanned Land Vehicle — ULV) NavLab, способное автономно передвигаться по автомагистралям в режиме самовожждения. К другим типам мобильных роботов относятся **автоматическое воздушное транспортное средство** (Unmanned Air Vehicle — UAV), обычно используемое для воздушного наблюдения, химической обработки земельных участков и военных операций, **автономное подводное транспортное средство** (Autonomous Underwater Vehicle — AUV), используемое в глубоководных морских исследованиях, и **планетоход**, такой как робот Sojourner, показанный на рис. 25.1, а.



а)



б)

Рис. 25.1. Фотографии широко известных роботов: движущийся робот Sojourner агентства NASA, который исследовал поверхность Марса в июле 1997 года (а); роботы-гуманоиды P3 и Asimo компании Honda (б)

К третьему типу относятся гибридные устройства — мобильные роботы, оборудованные манипуляторами. В их число входят **роботы-гуманоиды**, которые по своей физической конструкции напоминают человеческое тело. Два таких робота-гуманоида показаны на рис. 25.1, б; оба они изготовлены в японской корпорации Honda. Гибридные роботы способны распространить действие своих исполнительных элементов на более обширную рабочую область по сравнению с прикрепленными к одному месту манипуляторами, но вынуждены выполнять стоящие перед ними задачи с большими усилиями, поскольку не имеют такой жесткой опоры, которую предоставляет узел крепления манипулятора.

К сфере робототехники относятся также протезные устройства (искусственные конечности, ушные и глазные протезы для людей), интеллектуальные системы жиз-

необеспечения (например, целые дома, оборудованные датчиками и исполнительными механизмами), а также многотельные системы, в которых робототехнические действия осуществляются с использованием целых полчищ небольших роботов, объединяющих свои усилия.

Реальным роботам обычно приходится действовать в условиях среды, которая является частично наблюдаемой, стохастической, динамической и непрерывной. Некоторые варианты среды обитания роботов (но не все) являются также последовательными и мультиагентными. Частичная наблюдаемость и стохастичность обусловлены тем, что роботу приходится сталкиваться с большим, сложным миром. Робот не может заглянуть за каждый угол, а команды на выполнение движений осуществляются не с полной определенностью из-за проскальзывания приводных механизмов, трения и т.д. Кроме того, реальный мир упорно отказывается действовать быстрее, чем в реальном времени. В моделируемой среде предоставляется возможность использовать простые алгоритмы (такие как алгоритм **Q-обучения**, описанный в главе 21), чтобы определить с помощью обучения необходимые параметры, осуществляя миллионы попыток в течение всего лишь нескольких часов процессорного времени, а в реальной среде для выполнения всех этих попыток могут потребоваться годы. Кроме того, реальные аварии, в отличие от моделируемых, действительно наносят ущерб. В применяемые на практике робототехнические системы необходимо вносить априорные знания о роботе, о его физической среде и задачах, которые он должен выполнять для того, чтобы быстро пройти обучение и действовать безопасно.

25.2. АППАРАТНОЕ ОБЕСПЕЧЕНИЕ РОБОТОВ

До сих пор в этой книге предполагалось, что конструкция компонентов архитектуры агентов (датчиков, исполнительных механизмов и процессоров) уже определена и осталось лишь заняться разработкой программы агента. Но успехи в создании реальных роботов не в меньшей степени зависят от того, насколько удачно будут спроектированы датчики и исполнительные механизмы, подходящие для выполнения поставленной задачи.

Датчики

Датчики — это не что иное, как интерфейс между роботами и той средой, в которой они действуют, обеспечивающий передачу результатов восприятия. ➤ **Пассивные датчики**, такие как видеокамеры, в полном смысле этого слова выполняют функции наблюдателя за средой — они перехватывают сигналы, создаваемые другими источниками сигналов в среде. ➤ **Активные датчики**, такие как локаторы, посылают энергию в среду. Их действие основано на том, что часть излучаемой энергии отражается и снова поступает в датчик. Как правило, активные датчики позволяют получить больше информации, чем пассивные, но за счет увеличения потребления энергии от источника питания; еще одним их недостатком является то, что при одновременном использовании многочисленных активных датчиков может возникнуть интерференция. В целом датчики (активные и пассивные) можно разбить на три типа, в зависимости от того, регистрируют ли они расстояния до объектов, формируют изображения среды или контролируют характеристики самого робота.

В большинстве мобильных роботов используются **дальномеры**, которые представляют собой датчики, измеряющие расстояние до ближайших объектов. Одним из широко применяемых типов таких датчиков является **звуковой локаатор**, известный также как ультразвуковой измерительный преобразователь. Звуковые локаторы излучают направленные звуковые волны, которые отражаются от объектов, и часть этого звука снова поступает в датчик. При этом время поступления и интенсивность такого возвратного сигнала несут информацию о расстоянии до ближайших объектов. Для автономных подводных аппаратов преимущественно используется технология подводных гидролокаторов, а на земле звуковые локаторы в основном используются для предотвращения столкновений лишь в ближайших окрестностях, поскольку эти датчики характеризуются ограниченным угловым разрешением. К числу других устройств, альтернативных по отношению к звуковым локаторам, относятся радары (в основном применяемые на воздушных судах) и лазеры. Лазерный дальномер показан на рис. 25.2.

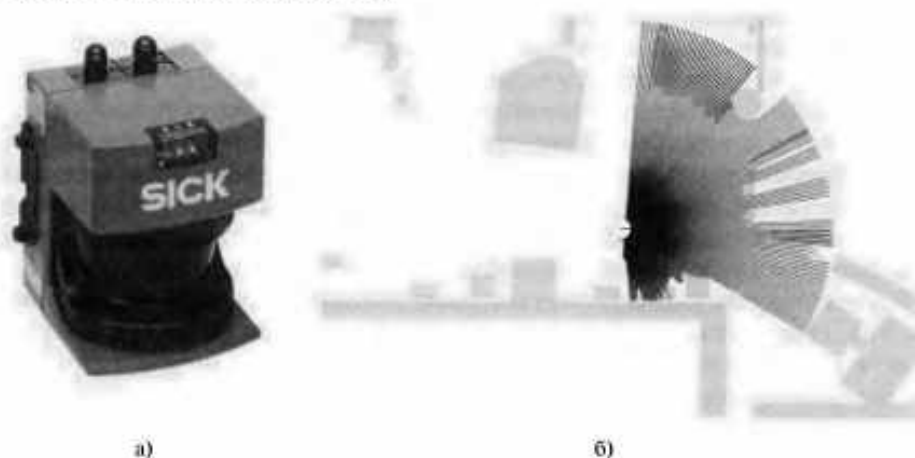


Рис. 25.2. Типичный пример датчика и его практического применения: лазерный дальномер (датчик расстояния) SICK LMS — широко применяемый датчик для мобильных роботов (а); результаты измерения расстояний, полученные с помощью горизонтально установленного датчика расстояния, спроектированные на двухмерную карту среды (б)

Некоторые датчики расстояния предназначены для измерения очень коротких или очень длинных расстояний. В число датчиков измерения коротких расстояний входят **тактильные датчики**, такие как контактные усики, контактные панели и сенсорные покрытия. На другом конце спектра находится **глобальная система позиционирования** (Global Positioning System — GPS), которая измеряет расстояние до спутников, излучающих импульсные сигналы. В настоящее время на орбите находятся свыше двух десятков спутников, каждый из которых передает сигналы на двух разных частотах. Приемники GPS определяют расстояние до этих спутников, анализируя значения фазовых сдвигов. Затем, выполняя триангуляцию сигналов от нескольких спутников, приемники GPS определяют свои абсолютные координаты на Земле с точностью до нескольких метров. В **дифференциальных системах GPS** применяется второй наземный приемник с известными координатами, благодаря чему при идеальных условиях обеспечивается точность измерения коор-

динат до миллиметра. К сожалению, системы GPS не работают внутри помещения или под водой.

Вторым важным классом датчиков являются **датчики изображения** — видеокамеры, позволяющие получать изображения окружающей среды, а также моделировать и определять характеристики среды с использованием методов машинного зрения, описанных в главе 24. В робототехнике особо важное значение имеет стереоскопическое зрение, поскольку оно позволяет получать информацию о глубине; тем не менее будущее этого направления находится под угрозой, поскольку успешно осуществляется разработка новых активных технологий получения пространственных изображений.

К третьему важному классу относятся **проприоцептивные датчики**, которые информируют робота о его собственном состоянии. Для измерения точной конфигурации робототехнического шарнира приводящие его в действие электродвигатели часто оснащаются **дешифраторами угла поворота вала**, которые позволяют определять даже небольшие приращения угла поворота вала электродвигателя. В манипуляторах роботов дешифраторы угла поворота вала способны предоставить точную информацию за любой период времени. В мобильных роботах дешифраторы угла поворота вала, которые передают данные о количестве оборотов колеса, могут использоваться для **одометрии** — измерения пройденного расстояния. К сожалению, колеса часто сдвигаются и проскальзывают, поэтому результаты одометрии являются точными только для очень коротких расстояний. Еще одной причиной ошибок при определении позиции являются внешние силы, такие как течения, воздействующие на автономные подводные аппараты, и ветры, сбивающие с курса автоматические воздушные транспортные средства. Улучшить эту ситуацию можно с использованием **инерционных датчиков**, таких как гироскопы, но даже они, применяемые без других дополнительных средств, не позволяют исключить неизбежное накопление погрешности определения положения робота.

Другие важные аспекты состояния робота контролируются с помощью **датчиков усилия** и **датчиков вращающего момента**. Без этих датчиков нельзя обойтись, если роботы предназначены для работы с хрупкими объектами или объектами, точная форма и местонахождение которых неизвестны. Представьте себе, что робототехнический манипулятор с максимальным усилием сжатия в одну тонну закручивает в патрон электрическую лампочку. При этом очень трудно предотвратить такую ситуацию, что робот приложит слишком большое усилие и раздавит лампочку. Но датчики усилия позволяют роботу ощутить, насколько крепко он держит лампочку, а датчики вращающего момента — определить, с каким усилием он ее поворачивает. Хорошие датчики позволяют измерять усилия в трех направлениях переноса и трех направлениях вращения.

Исполнительные механизмы

Исполнительные механизмы являются теми средствами, с помощью которых роботы передвигаются и изменяют форму своего тела. Для того чтобы понять основные особенности конструкции исполнительных механизмов, необходимо вначале рассмотреть абстрактные понятия движения и формы, используя концепцию **степени свободы**. Как степень свободы мы будем рассматривать каждое независимое направление, в котором могут передвигаться либо робот, либо один из его

исполнительных механизмов. Например, твердотельный свободно движущийся робот, такой как автономный подводный аппарат, имеет шесть степеней свободы; три из них, (x, y, z) , определяют положение робота в пространстве, а три других — его угловую ориентацию по трем осям вращения, известную как качание (yaw), поворот (roll) и наклон (pitch). Эти шесть степеней свободы определяют **кинематическое состояние**² или **позу** робота. **Динамическое состояние** робота включает по одному дополнительному измерению для скорости изменения каждого кинематического измерения.

Роботы, не являющиеся твердотельными, имеют дополнительные степени свободы внутри самих себя. Например, в руке человека локоть имеет одну степень свободы (может сгибаться в одном направлении), а кисть имеет три степени свободы (может двигаться вверх и вниз, из стороны в сторону, а также вращаться). Каждый из шарниров робота также имеет 1, 2 или 3 степени свободы. Для перемещения любого объекта, такого как рука, в конкретную точку с конкретной ориентацией необходимо иметь шесть степеней свободы. Рука, показанная на рис. 25.3, а, имеет точно шесть степеней свободы, создаваемых с помощью пяти **поворотных шарниров**, которые формируют вращательное движение, и одного **призматического сочленения**, который формирует скользящее движение. Чтобы убедиться в том, что рука человека в целом имеет больше шести степеней свободы, можно провести простой эксперимент: положите кисть на стол и убедитесь в том, что вы еще имеете возможность поворачивать руку в локте, не меняя положения кисти на столе. Манипуляторами, имеющими больше степеней свободы, чем требуется для перевода конечного исполнительного механизма в целевое положение, проще управлять по сравнению с роботами, имеющими лишь минимальное количество степеней свободы.

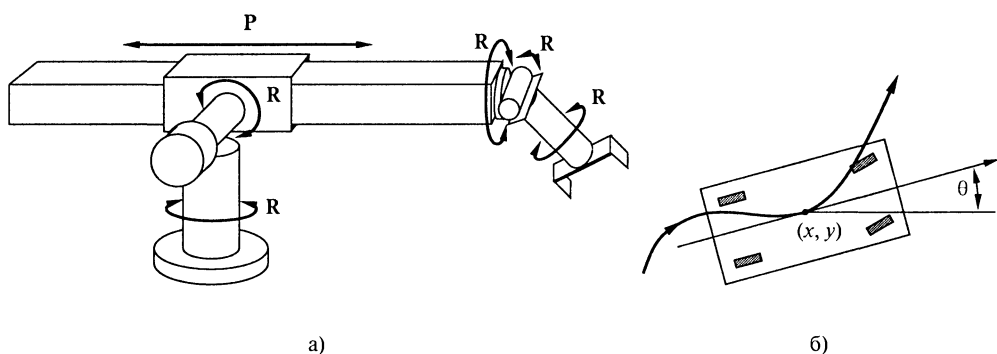


Рис. 25.3. Особенности конструкции манипулятора робота: станфордский манипулятор (Stanford Manipulator) — один из первых манипуляторов робота, в котором используются пять поворотных шарниров (R) и одно призматическое сочленение (P), что позволяет получить в целом шесть степеней свободы (а); траектория движения неголономного четырехколесного транспортного средства с рулевым управлением от передних колес (б)

В мобильных роботах количество степеней свободы не обязательно совпадает с количеством приводимых в действие элементов. Рассмотрим, например, обычный автомобиль: он может передвигаться вперед или назад, а также поворачиваться, что

² Термин “кинематика”, как и слово “кинематограф”, происходит от греческого корня, обозначающего движение.

соответствует двум степеням свободы. В отличие от этого кинематическая конфигурация автомобиля является трехмерной — на открытой плоской поверхности можно легко перевести автомобиль в любую точку (x, y) , с любой ориентацией (см. рис. 25.3, б). Таким образом, автомобиль имеет три **эффективные степени свободы**, но две **управляемые степени свободы**. Робот называется **неголономным**, если он имеет больше эффективных степеней свободы, чем управляемых степеней свободы, и **голономным**, если эти два значения совпадают. Голономные роботы проще в управлении (было бы намного легче припарковать автомобиль, способный двигаться не только вперед и назад, но и в стороны), однако голономные роботы являются также механически более сложными. Большинство манипуляторов роботов являются голономными, а большинство мобильных роботов — неголономными.

В мобильных роботах применяется целый ряд механизмов для перемещения в пространстве, включая колеса, гусеницы и ноги. Роботы с **дифференциальным приводом** оборудованы расположенными с двух сторон независимо активизируемыми колесами (или гусеницами, как в армейском танке). Если колеса, находящиеся с обеих сторон, вращаются с одинаковой скоростью, то робот движется по прямой. Если же они вращаются в противоположных направлениях, то робот поворачивается на месте. Альтернативный вариант состоит в использовании **синхронного привода**, в котором каждое колесо может вращаться и поворачиваться вокруг вертикальной оси. Применение такой системы привода вполне могло бы привести к хаотическому перемещению, если бы не использовалось такое ограничение, что все пары колес поворачиваются в одном направлении и вращаются с одинаковой скоростью. И дифференциальный, и синхронный приводы являются неголономными. В некоторых более дорогостоящих роботах используются голономные приводы, которые обычно состоят из трех или большего количества колес, способных поворачиваться и вращаться независимо друг от друга.

Ноги, в отличие от колес, могут использоваться для передвижения не по плоской поверхности, а по местности, характеризующейся очень грубым рельефом. Тем не менее на плоских поверхностях ноги как средства передвижения значительно уступают колесам, к тому же задача создания для них механической конструкции является очень сложной. Исследователи в области робототехники предприняли попытки разработать конструкции с самым разным количеством ног, начиная от одной ноги и заканчивая буквально десятками. Были разработаны роботы, оборудованные ногами для ходьбы, бега и даже прыжков (как показано на примере шагающего робота на рис. 25.4, а). Этот робот является **динамически устойчивым**; это означает, что он может оставаться в вертикальном положении, только непрерывно двигаясь. Робот, способный оставаться в вертикальном положении, не двигая ногами, называется **статически устойчивым**. Робот является статически устойчивым, если центр его тяжести находится над многоугольником, охваченным его ногами.

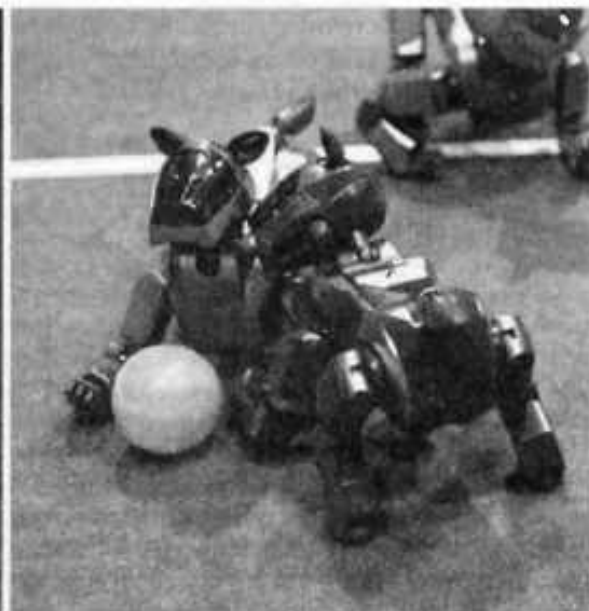
В мобильных роботах других типов для передвижения используются иные, чрезвычайно разнообразные механизмы. В летательных аппаратах обычно применяются пропеллеры или турбины. Роботизированные дирижабли держатся в воздухе за счет тепловых эффектов. В автономных подводных транспортных средствах часто используются подруливающие устройства, подобные тем, которые устанавливаются на подводных лодках.

Для того чтобы робот мог функционировать, ему недостаточно быть оборудованным только датчиками и исполнительными механизмами. Полноценный робот дол-

жен также иметь источник энергии для привода своих исполнительных механизмов. Для приведения в действие манипулятора и для передвижения чаще всего используются **электродвигатели**; определенную область применения имеют также **пневматические приводы**, в которых используется сжатый газ, и **гидравлические приводы**, в которых используется жидкость под высоким давлением. Кроме того, в большинстве роботов имеются некоторые средства цифровой связи наподобие беспроводной сети. Наконец, робот должен иметь жесткий корпус, на который можно было бы навесить все эти устройства, а также, фигурально выражаясь, держать при себе паяльник, на тот случай, что его оборудование перестанет работать.



а)



б)

Рис. 25.4. Примеры роботов, передвигающихся с помощью ног: один из шагающих роботов Марка Рэйберта (Marc Raibert) в движении (а); роботы AIBO компании Sony, играющие в футбол (© от 2001 года, федерация RoboCup) (б)

25.3. ВОСПРИЯТИЕ, ОСУЩЕСТВЛЯЕМОЕ РОБОТАМИ

Робототехническое восприятие — это процесс, в ходе которого роботы отображают результаты сенсорных измерений на внутренние структуры представления среды. Задача восприятия является сложной, поскольку информация, поступающая от датчиков, как правило, зашумлена, а среда является частично наблюдаемой, непредсказуемой и часто динамической. В качестве эмпирического правила можно руководствоваться тем, что качественные внутренние структуры представления обладают тремя свойствами: содержат достаточно информации для того, чтобы робот мог принимать правильные решения, построены так, чтобы их можно было эффективно обновлять, и являются естественными в том смысле, что внутренние переменные соответствуют естественным переменным состояниям в физическом мире.

В главе 15 было показано, что модели перехода и восприятия для частично наблюдаемой среды могут быть представлены с помощью фильтров Калмана, скрытых марковских моделей и динамических байесовских сетей; кроме того, в указанной главе были описаны и точные, и приближенные алгоритмы обновления **доверительного состояния** — распределения апостериорных вероятностей по переменным состояния среды. К тому же в главе 15 было приведено несколько динамических моделей байесовских сетей для этого процесса. А при решении робототехнических задач в модель в качестве наблюдаемых переменных обычно включают собственные прошлые действия робота (пример такой сети см. на рис. 17.7). На рис. 25.5 показана система обозначений, используемая в данной главе: \mathbf{X}_t — это состояние среды (включая робота) во время t ; \mathbf{Z}_t — результаты наблюдений, полученные во время t ; \mathbf{A}_t — действие, предпринятое после получения этих результатов наблюдения.

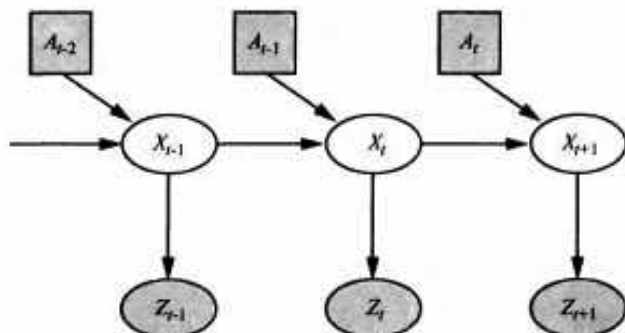


Рис. 25.5. Процесс робототехнического восприятия, рассматриваемый как временной алгоритмический вывод на основании последовательностей действий и измерений, который демонстрируется на примере динамической байесовской сети

Задача **фильтрации**, или обновления доверительного состояния, по сути является такой же, как и в главе 15. Эта задача состоит в том, что должно быть вычислено новое доверительное состояние $P(\mathbf{X}_{t+1} | \mathbf{z}_{1:t+1}, \mathbf{a}_{1:t})$ на основании текущего доверительного состояния $P(\mathbf{X}_t | \mathbf{z}_{1:t}, \mathbf{a}_{1:t-1})$ и нового наблюдения \mathbf{z}_{t+1} . Принципиальные различия по сравнению с указанной главой состоят в следующем: во-первых, результаты вычислений явно обусловлены не только действиями, но и наблюдениями, и, во-вторых, теперь приходится иметь дело с непрерывными, а не с дискретными переменными. Таким образом, необходимо следующим образом откорректировать рекурсивное уравнение фильтрации (15.3) для использования в нем интегрирования, а не суммирования:

$$P(\mathbf{X}_{t+1} | \mathbf{z}_{1:t+1}, \mathbf{a}_{1:t})$$

$$= \alpha P(\mathbf{z}_{t+1} | \mathbf{X}_{t+1}) \int P(\mathbf{X}_{t+1} | \mathbf{X}_t, \mathbf{a}_t) P(\mathbf{X}_t | \mathbf{z}_{1:t}, \mathbf{a}_{1:t-1}) d\mathbf{X}_t \quad (25.1)$$

Это уравнение показывает, что апостериорное распределение вероятностей по переменным состояния \mathbf{X} во время $t+1$ вычисляется рекурсивно на основании соответствующей оценки, полученной на один временной шаг раньше.

В этих вычислениях участвуют данные о предыдущем действии a_t и о текущих сенсорных измерениях z_{t+1} . Например, если цель заключается в разработке робота, играющего в футбол, то X_{t+1} может представлять местонахождение футбольного мяча относительно робота. Распределение апостериорных вероятностей $P(X_t | z_{1:t}, a_{1:t-1})$ — это распределение вероятностей по всем состояниям, отражающее все, что известно о прошлых результатах сенсорных измерений и об управляющих воздействиях. Уравнение 25.1 показывает, как рекурсивно оценить это местонахождение, инкрементно развертывая вычисления и включая в этот процесс данные сенсорных измерений (например, изображения с видеокamerы) и команды управления движением робота. Вероятность $P(X_{t+1} | x_t, a_t)$ называется **моделью перехода**, или **моделью движения**, а вероятность $P(z_{t+1} | X_{t+1})$ представляет собой **модель восприятия**.

Локализация

🔗 **Локализация** — это универсальный пример робототехнического восприятия. Она представляет собой задачу определения того, где что находится. Локализация — одна из наиболее распространенных задач восприятия в робототехнике, поскольку знания о местонахождении объектов и самого действующего субъекта являются основой любого успешного физического взаимодействия. Например, роботы, относящиеся к типу манипуляторов, должны иметь информацию о местонахождении объектов, которыми они манипулируют. А роботы, передвигающиеся в пространстве, должны определять, где находятся они сами, чтобы прокладывать путь к целевым местонахождениям.

Существуют три разновидности задачи локализации с возрастающей сложностью. Если первоначальная поза локализуемого объекта известна, то локализация сводится к задаче 🔗 **отслеживания траектории**. Задачи отслеживания траектории характеризуются ограниченной неопределенностью. Более сложной является задача 🔗 **глобальной локализации**, в которой первоначальное местонахождение объекта полностью неизвестно. Задачи глобальной локализации преобразуются в задачи отслеживания траектории сразу после локализации искомого объекта, но в процессе их решения возникают также такие этапы, когда роботу приходится учитывать очень широкий перечень неопределенных состояний. Наконец, обстоятельства могут сыграть с роботом злую шутку и произойдет “похищение” (т.е. внезапное исчезновение) объекта, который он пытался локализовать. Задача локализации в таких неопределенных обстоятельствах называется 🔗 **задачей похищения**. Ситуация похищения часто используется для проверки надежности метода локализации в крайне неблагоприятных условиях.

В целях упрощения предположим, что робот медленно движется на плоскости и что ему дана точная карта среды (пример подобной карты показан на рис. 25.7). Поза такого мобильного робота определяется двумя декартовыми координатами со значениями x и y , а также его угловым направлением со значением θ , как показано на рис. 25.6, а. (Обратите внимание на то, что исключены соответствующие скорости, поэтому рассматриваемая модель скорее является кинематической, а не динамической.) Если эти три значения будут упорядочены в виде вектора, то любое конкретное состояние определится с помощью соотношения $X_t = (x_t, y_t, \theta_t)^T$.

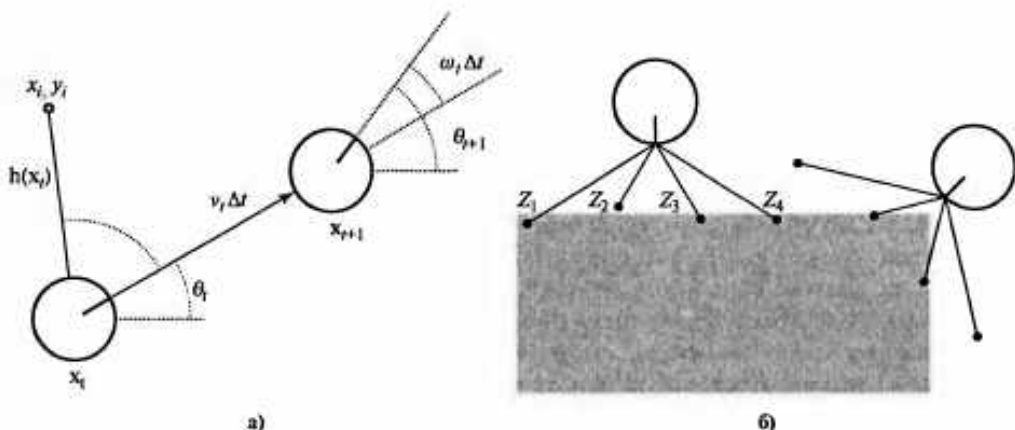


Рис. 25.6. Пример применения карты среды: упрощенная кинематическая модель мобильного робота. Робот показан в виде кружка с отметкой, обозначающей переднее направление. Показаны значения позиции и ориентации в моменты времени t и $t+1$, а обновления обозначены соответственно терминами $v_t \Delta t$ и $\omega_t \Delta t$. Кроме того, приведена отметка с координатой (x_i, y_i) , наблюдаемая во время t (а); модель датчика расстояния. Показаны две позы робота, соответствующие заданным результатам измерения расстояний (z_1, z_2, z_3, z_4) . Гораздо более вероятным является предположение, что эти результаты измерения расстояний получены в позе, показанной слева, а не справа (б)

В этой кинематической аппроксимации каждое действие состоит из “мгновенной” спецификации двух скоростей — скорости переноса v_t и скорости вращения ω_t . Для небольших временных интервалов Δt грубая детерминированная модель движения таких роботов задается следующим образом:

$$\hat{\mathbf{x}}_{t+1} = f(\mathbf{x}_t, \underbrace{v_t, \omega_t}_{\mathbf{a}_t}) = \mathbf{x}_t + \begin{pmatrix} v_t \Delta t \cos \theta_t \\ v_t \Delta t \sin \theta_t \\ \omega_t \Delta t \end{pmatrix}$$

Обозначение $\hat{\mathbf{x}}$ относится к детерминированному предсказанию состояния. Безусловно, поведение физических роботов является довольно непредсказуемым. Такая ситуация обычно моделируется гауссовым распределением со средним $f(\mathbf{x}_t, v_t, \omega_t)$ и ковариацией Σ_x (математическое определение приведено в приложении А).

$$\mathbf{P}(\mathbf{x}_{t+1} | \mathbf{x}_t, v_t, \omega_t) = N(\hat{\mathbf{x}}_{t+1}, \Sigma_x)$$

Затем необходимо разработать модель восприятия. Рассмотрим модели восприятия двух типов. В первой из них предполагается, что датчики обнаруживают стабильные, различимые характеристики среды, называемые **отметками**. Для каждой отметки они сообщают дальность и азимут. Предположим, что состояние робота определяется выражением $\mathbf{x}_t = (x_t, y_t, \theta_t)^T$ и он принимает информацию об отметке, местонахождение которой, как известно, определяется координатами $(x_i, y_i)^T$. При отсутствии шума дальность и азимут можно вычислить с помощью простого геометрического соотношения (см. рис. 25.6, а). Точное предсказание наблюдаемых значений дальности и азимута может быть выполнено с помощью следующей формулы:

$$\hat{\mathbf{z}}_t = h(\mathbf{x}_t) = \begin{pmatrix} \sqrt{(x_t - x_i)^2 + (y_t - y_i)^2} \\ \arctan \frac{y_i - y_t}{x_i - x_t} - \theta_t \end{pmatrix}$$

Еще раз отметим, что полученные результаты измерений искажены шумом. Для упрощения можно предположить наличие гауссова шума с ковариацией Σ_z :

$$P(\mathbf{z}_t | \mathbf{x}_t) = N(\hat{\mathbf{z}}_t, \Sigma_z)$$

Для дальномеров такого типа, как показаны на рис. 25.2, часто более приемлемой является немного другая модель восприятия. Такие датчики вырабатывают вектор значений дальности $\mathbf{z}_t = (z_1, \dots, z_M)^T$, в каждом из которых азимуты являются фиксированными по отношению к роботу. При условии, что дана поза \mathbf{x}_t , допустим, что z_j — точное расстояние вдоль направления j -го луча от \mathbf{x}_t до ближайшего препятствия. Как и в описанном раньше случае, эти результаты могут быть искажены гауссовым шумом. Как правило, предполагается, что погрешности для различных направлений лучей независимы и заданы в виде идентичных распределений, поэтому имеет место следующая формула:

$$P(\mathbf{z}_t | \mathbf{x}_t) = \alpha \prod_{j=1}^M e^{-(z_j - \hat{z}_j) / 2\sigma^2}$$

На рис. 25.6, б показан пример четырехлучевого дальмера и двух возможных поз робота, одну из которых на полном основании можно рассматривать как позу, в которой были получены рассматриваемые результаты измерения дальностей, а другую — нет. Сравнивая модель измерения дальностей с моделью отметок, можно убедиться в том, что модель измерения дальностей обладает преимуществом в том, что не требует идентификации отметки для получения возможности интерпретировать результаты измерения дальностей; и действительно, как показано на рис. 25.6, б, робот направлен в сторону стены, не имеющей характерных особенностей. С другой стороны, если бы перед ним была видимая, четко идентифицируемая отметка, то робот мог бы обеспечить немедленную локализацию.

В главе 15 описаны фильтр Калмана, позволяющий представить доверительное состояние в виде одного многомерного гауссова распределения, и фильтр частиц, который представляет доверительное состояние в виде коллекций частиц, соответствующих состоянию. В большинстве современных алгоритмов локализации используется одно из этих двух представлений доверительного состояния робота, $P(\mathbf{x}_t | \mathbf{z}_{1:t}, \mathbf{a}_{1:t-1})$.

Локализация с использованием фильтрации частиц называется **локализацией Монте-Карло**, или сокращенно MCL (Monte Carlo Localization). Алгоритм MCL идентичен алгоритму фильтрации частиц, приведенному в листинге 15.3; достаточно лишь предоставить подходящую модель движения и модель восприятия. Одна из версий алгоритма, в которой используется модель измерения дальностей, приведена в листинге 25.1. Работа этого алгоритма продемонстрирована на рис. 25.7, где показано, как робот определяет свое местонахождение в офисном здании. На первом изображении частицы распределены равномерно согласно распределению априорных вероятностей, показывающему наличие глобальной неопределенности в отношении положения робота. На втором изображении показано, как поступает первый

ряд результатов измерений и частицы формируют кластеры в областях с высоким распределением апостериорных доверительных состояний. А на третьем изображении показано, что поступило достаточное количество результатов измерений, чтобы переместить все частицы в одно место.

Листинг 25.1. Алгоритм локализации Монте-Карло, в котором используется модель восприятия результатов измерения дальностей с учетом наличия независимого шума

```

function Monte-Carlo-Localization( $a, z, N, model, map$ ) returns множество
    выборок  $S$ 
inputs:  $a$ , предыдущая команда приведения робота в движение
            $z$ , результаты измерения дальностей с  $M$  отсчетами  $z_1, \dots, z_M$ 
            $N$ , количество сопровождаемых выборок
            $model$ , вероятностная модель среды с данными о предыдущей
               позе  $\mathbf{P}(\mathbf{X}_0)$ , моделью движения  $\mathbf{P}(\mathbf{X}_1 | \mathbf{X}_0, A_0)$  и моделью
               шума для датчика расстояний  $P(Z | \hat{Z})$ 
            $map$ , двумерная карта среды
static:  $S$ , вектор выборок с размером  $N$ , первоначально вырабатываемый
           из  $\mathbf{P}(\mathbf{X}_0)$ 
local variables:  $W$ , вектор весов с размером  $N$ 

for  $i = 1$  to  $N$  do
     $S[i] \leftarrow$  выборка из  $\mathbf{P}(\mathbf{X}_1 | \mathbf{X}_0 = S[i], A_0 = a)$ 
     $W[i] \leftarrow 1$ 
    for  $j = 1$  to  $M$  do
         $\hat{z} \leftarrow \text{Exact-Range}(j, S[i], map)$ 
         $W[i] \leftarrow W[i] \cdot P(Z = z_j | \hat{Z} = \hat{z})$ 
 $S \leftarrow \text{Weighted-Sample-With-Replacement}(N, S, W)$ 
return  $S$ 

```

Еще один важный способ локализации основан на применении фильтра Калмана. Фильтр Калмана представляет апостериорную вероятность $\mathbf{P}(\mathbf{X}_t | \mathbf{z}_{1:t}, a_{1:t-1})$ с помощью гауссова распределения. Среднее этого гауссова распределения будет обозначено μ_t , а его ковариация — Σ_t . Основным недостатком использования гауссовых доверительных состояний является то, что они замкнуты только при использовании линейных моделей движения f и линейных моделей измерения h . В случае нелинейных f или h результат обновления фильтра обычно не является гауссовым. Таким образом, алгоритмы локализации, в которых используется фильтр Калмана, ~~линеаризуют~~ модели движения и восприятия. *Линеаризацией* называется локальная аппроксимация нелинейной функции с помощью линейной.