**Слайд 1.**

**Слайд 2.**

HOG — дескрипторы особых точек, которые используются в компьютерном зрении и обработке изображений с целью распознавания объектов. Данная техника основана на подсчете количества направлений градиента в локальных областях изображения.

Распознавание лиц представляет собой фактически последовательность нескольких связанных проблем:  
1. Во-первых, необходимо рассмотреть изображение и найти на нём все лица.  
2. Во-вторых, необходимо сосредоточиться на каждом лице и определить, что, несмотря на неестественный поворот лица или неважное освещение, это — один и тот же человек.  
3. В-третьих, надо выделить уникальные характеристики лица, которые можно использовать для отличия его от других людей — например, размер глаз, удлинённость лица и т.п.  
4. В завершение необходимо сравнить эти уникальные характеристики лица с характеристиками других известных вам людей, чтобы определить имя человека.  
Мозг человека проделывает всё это автоматически и мгновенно.

**Слайд 3**

Обнаружение лица стало господствующей тенденцией в начале 2000-х годов, когда Пол Виола и Майкл Джонс изобрели способ обнаруживать лица, который был достаточно быстрым, чтобы работать на дешёвых камерах. Однако сейчас существуют намного более надёжные решения. Мы собираемся использовать метод, открытый в 2005 году, — гистограмма направленных градиентов (коротко, **HOG**).

**Слайд 4**

Для обнаружения лиц на изображении мы сделаем наше изображение чёрно-белым, т.к. данные о цвете не нужны для обнаружения лиц:

**Слайд 5**

Затем мы рассмотрим каждый отдельный пиксель на нашем изображении последовательно. Для каждого отдельного пикселя следует рассмотреть его непосредственное окружение:

**Слайд 6**

Нашей целью является выделить, насколько тёмным является текущий пиксель по сравнению с пикселями, прямо примыкающими к нему. Затем проведём стрелку, показывающую направление, в котором изображение становится темнее:

При рассмотрении этого одного пикселя и его ближайших соседей видно, что изображение темнеет вверх вправо.  
Если повторить этот процесс для **каждого отдельного пикселя** на изображении, то, в конечном итоге, каждый пиксель будет заменён стрелкой. Эти стрелки называют градиентом, и они показывают поток от света к темноте по всему изображению:

Может показаться, что результатом является нечто случайное, но есть очень хорошая причина для замены пикселей градиентами. Когда мы анализируем пиксели непосредственно, то у тёмных и светлых изображений одного и того же человека будут сильно различающиеся значения интенсивности пикселей. Но если рассматривать только направление изменения яркости, то как тёмное, так и светлое изображения будут иметь совершенно одинаковое представление. Это значительно облегчает решение проблемы!  
Но сохранение градиента для каждого отдельного пикселя даёт нам способ, несущий слишком много подробностей. Было бы лучше, если бы мы могли просто видеть основной поток светлого/тёмного на более высоком уровне, рассматривая таким образом базовую структуру изображения.  
**Слайд 7**  
Для этого разбиваем изображение на небольшие квадраты 16х16 пикселей в каждом. В каждом квадрате следует подсчитать, сколько градиентных стрелок показывает в каждом главном направлении (т.е. сколько стрелок направлено вверх, вверх-вправо, вправо и т.д.). Затем рассматриваемый квадрат на изображении заменяют стрелкой с направлением, преобладающим в этом квадрате.  
В конечном результате мы превращаем исходное изображение в очень простое представление, которое показывает базовую структуру лица в простой форме:  
**Слайд 8**  
Исходное изображение преобразовано в HOG-представление, демонстрирующее основные характеристики изображения независимо от его яркости.  
**Слайд 9**  
Чтобы обнаружить лица на этом HOG-изображении, всё, что требуется от нас, это найти такой участок изображения, который наиболее похож на известную HOG-структуру, полученную из группы лиц, использованной для обучения:

**Слайд 10**

Если есть желание выполнить этот этап самостоятельно, используя Python и dlib, то имеется программа, показывающая, как создавать и просматривать HOG-представления изображений.

https://gist.github.com/ageitgey/1c1cb1c60ace321868f7410d48c228e1