

# CGMIRT 2013 - Вопросы для зачёта

Денис Анисимов

22 декабря 2013 г.

## 1

*Каковы преимущества векторного представления изображения перед растровым представлением? Каковы преимущества растрового представления изображения перед векторным представлением?*

- Векторное представление - представление изображений геометрическими примитивами. Преимущества:
  1. Компактность представления. Для изображений, которые в основном представляют собой набор геометрических примитивов.
  2. Масштабируемость. Векторное представление описывает способ создания изображения (векторный путь), система координат которой при отрисовке может быть произвольно растянута.  
Одно и то же векторное представление может быть преобразовано в растровый с различным разрешением.
  3. Легче использование в векторных устройствах вывода, например плоттерах.
  4. Хорошо подходит для сжатия без потерь, т.к. являются, в основном, текстовым описанием векторов.
- Растровое представление - изображение в виде набора точек, обычно таблица в прямоугольной системе координат. Преимущества:
  - Распространённые устройства вывода (мониторы, принтеры) являются растровыми устройствами. Поэтому вывод растрового изображения (если не учитывать масштабирование и обработку):
    1. Быстрее - нет необходимости делать математические вычисления пути вектора.
    2. Точнее - результирующее изображение не зависит от алгоритма преобразования и на всех устройствах вывода выглядит одинаково. Нет артефактов, таких как алиасинг.
    3. Растровое представление позволяет использовать особенности вывода изображений на растровых устройствах. Пример - хинтинг шрифтов.
  - Для фотографических изображений растровое представление даёт большую точность при меньших накладных расходах. (сложные распределения пикселей трудны для представления в векторе и могут быть отрисованы с артефактами)

- Устройства ввода(камеры, сканеры) считывают растровое изображение. Перевод таких изображений в вектор процесс недетерминированный и не всегда практичный(см. предыдущий пункт).

## 2

*Как определяется пространство CIE XYZ. Какие свойства человеческого зрения делают это возможным?*

В человеческом глазе за восприятие цвета отвечают цветочувствительные клетки называемые "колбочки". Существует три типа колбочек, которые грубо можно обозначить как отвечающие за голубой(Short, 420-440нм), зелёный(Middle, 530-540нм) и красный(или оранжевый, Long, 560-580нм) цвета. Отклик этих клеток не монохроматический и поэтому в стандартных примитивах цвета RGB некоторые видимые глазом цвета могут соответствовать отрицательным значениям веса примитива. Чтобы избежать наличия отрицательных значений были введены мнимые основные цвета XYZ.

Другая особенность человеческого глаза - восприятие зелёного цвета как более яркого относительно синего и красного. Поэтому в XYZ модели Y считается координатой яркости и чувствительность яркости стандартного наблюдателя в модели CIE примерно соответствует отклику зелёных колбочек.

Три стандартных цвета XYZ дают трёхмерное пространство. Но для восприятия цвета человеческим глазом важно только относительные различия компонент, а абсолютные значения отвечают за яркость. Производная модель  $xuY$  использует относительные компоненты  $x$  и  $u$  и яркость  $Y$ . Цветовое пространство  $xu$  может быть представлено на плоскости в виде диаграммы-лепестка(форма следует из цветового восприятия стандартного наблюдателя).

## 3

*Пересечение кривой Безье и прямой линии. Быстрая оценка максимального количества пересечений. Алгоритм поиска пересечений.*

- Быструю и грубую оценку количества пересечений можно получить посчитав пересечения прямой с фигурой, образованной линией соединяющей контрольные точки кривой.
- Более точно оценить можно представив кривую Безье ломаной. Например последовательно деля её пополам.

(И первый и второй способ используют свойство кривой Безье, что она всегда лежит внутри выпуклой оболочки контрольных точек.)

- Точное нахождение пересечения прямой с кривой Безье можно получить математически. Представить прямую уравнением от координаты  $(x, y)$ , кривую параметрическим уравнением. Подставить значения координат вектора из второго в первое и решить уравнение с  $n$  корнями, где  $n$  - порядок кривой Безье. Можно решить символично(просто для линейных, квадратичных и кубических кривых) или приближённо, используя методы нахождения корней уравнения.

## 4

*Предсказание (экстраполяция, predictive coding) . Примеры использования в PNG и в JPEG.*

Экстраполяция - определение значения пиксела на основе уже известного. Кодировается не абсолютное значение, а его разница с предыдущим. В качестве предыдущего пиксела могут быть использованы несколько не обязательно смежных. Улучшение сжатия достигается за счёт того, что кодируемые значения имеют меньшую амплитуду и как правило постоянны (смежные пиксели в реальных изображениях редко существенно отличаются).

В сжатии без потерь JPEG-LS используется предсказание по одному, двум или трём непосредственно соседним пикселям. (кодирование изображения происходит при проходе сверху-вниз слева-направо, поэтому эти пиксели обозначаются как север, запад и северо-запад). В формате PNG также используется предсказание по алгоритму Паета, в котором из трёх пикселей выбирается ближайший к значению *север+запад - северо-запад*.

У такого кодирования разности есть недостаток - для восстановления значения в какой-то точке нужно просуммировать все разности от начального значения, т.е. кодирование не локально. Чтобы избежать этого можно применять кодирование Хаара, в котором вместе с полуразностью значений пикселей хранится их полусумма (сжатие за счёт того, что полусумма в два раза меньше и полуразности имеют малую амплитуду).

## 5

*Сравнить алгоритм буфера глубины и алгоритм ray-casting. В чем разница? Какие есть преимущества у алгоритма ray-casting?*

Буфер глубины и трассировка лучей - алгоритмы удаления невидимых поверхностей.

- Алгоритм буфера глубины использует буфер, аналогичный буферу изображения, для сохранения информации о глубине точки (расстоянию от плоскости изображения). При обработке объекта для каждой точки вычисляется её глубина, и если она меньше текущей, то значение точки в буфере изображения обновляется.
- Алгоритм трассировки лучей основывается на физической идее распространения света источника, отражённого от объекта, к наблюдателю. Обратная трассировка обозначает рассмотрение лучей доходящих до наблюдателя, вместо их распространения из источника. Наблюдателя можно считать бесконечно удалённым от плоскости изображения, и каждый луч проходящий через точку изображения можно считать параллельным остальным. Для каждого такого луча считаются пересечения с объектами, сортируются по глубине, и для задачи удаления невидимых плоскостей берётся ближайшее к плоскости изображения пересечение.

Трассировка лучей имеет ряд преимуществ перед буфером глубины:

- Проще распараллелить, т.к. каждый луч можно обрабатывать независимо.
- Меньше затраты по памяти, т.к. не надо хранить буфер глубины.
- Проще реализовать алгоритмы сглаживания и учёта прозрачности, т.к. информация от каждого объекта для каждого пиксела строго упорядочена.