

ВВЕДЕНИЕ В ZBrush® 4

Эрик Келлер



УДК 004.92ZBrush
ББК 32.973.26-018.2
К34

Келлер Э.
К34 Введение в ZBrush® 4. – М.: ДМК Пресс, 2012. – 768 с.: ил.

ISBN 978-5-94074-794-9

С этой книгой вы научитесь моделированию удивительных по своей реалистичности существ, людей и неодушевленных предметов в одной из лучших программ «цифровой лепки» Zbrush. Вы освоите уникальную технологию скульптинга, инновационный интерфейс и мощный набор инструментов Zbrush, работу с «цифровой глиной» и богатым арсеналом скульптурных кистей, моделирование скелетов и сеток при помощи Z-сфер и Z-скетчей, новые возможности создания объектов с твердой поверхностью, управление освещением, материалами и визуализацией, анимационные возможности ZBrush.

Издание предназначено для начинающих пользователей программы, но может принести неоценимую пользу и профессионалам, давно знакомым с Zbrush.

УДК 004.92ZBrush
ББК 32.973.26-018.2

Все права защищены. Любая часть этой книги не может быть воспроизведена в какой бы то ни было форме и какими бы то ни было средствами без письменного разрешения владельцев авторских прав.

Материал, изложенный в данной книге, многократно проверен. Но поскольку вероятность технических ошибок все равно существует, издательство не может гарантировать абсолютную точность и правильность приводимых сведений. В связи с этим издательство не несет ответственности за возможные ошибки, связанные с использованием книги.

ISBN 978-0-470-52764-1 (англ.)

Copyright © 2011 by Wiley
Publishing, Inc.

ISBN 978-5-94074-794-9 (рус.)

© Перевод на русский язык,
оформление, ДМК Пресс, 2012



Содержание

Благодарности	18
Об авторе	19
Предисловие	20
Вступление	22
Для кого предназначена эта книга	24
Что вы найдете внутри этой книги	25
Как связаться с автором книги	27
Глава 1. Основы компьютерной графики	28
Введение в ZBrush	29
Понятие о цифровом изображении	31
Анатомия пиксела	31
Сглаживание краев изображения	31
Каналы и глубина цвета	33
Форматы файлов изображений	35
Векторные изображения	37
Что такое «разрешение изображения»	37
Разрешение экрана	38
Разрешение документа	39
Разрешение изображения	40
Соотношение размеров	40
Полигональное разрешение	41
Понятие о трехмерном пространстве	42
Анатомия полигона	43
Пиксолы и пикселы	46
Что значит быть цифровым художником	47
Ресурсы	48

Глава 2. Лицом к лицу с интерфейсом	51
Философия ZBrush	52
Холст ZBrush	53
Light Box	54
Панели инструментов ZBrush	57
Левая панель	57
Верхняя панель	63
Как работает режим правки (Edit)	64
Правая панель	71
Доки и палитры	78
Заглавное меню	101
Резюме	102
Глава 3. Основы «цифровой лепки»	103
«Цифровая глина»	104
Загружаем полисферу	106
Кисти Standard, Smooth и Move	113
Кисть Standard	114
Кисть Smooth	122
Кисть Move	125
Увеличиваем плотность сетки	127
Используем функцию симметрии	130
Моделируем голову дракона	134
Поговорим о формах	135
Подготовка к работе	135
Кисть Move Elastic	137
Кисть Move Topological	143
Уточняем базовую форму головы дракона	145
Кисть Clay	145
Кисть Clay Build Up	148
Маскирование	153
Управление процессом маскирования	153
Полигруппы	162
Выделение полигонов	163

Создание полигруппы из выделенной области	168
Создание полигруппы из маски.....	170
Создаем полигруппу из полигонов нижней челюсти дракона.....	171
Создаем геометрию с помощью краевых петель	175
Открываем дракону пасть с помощью транспозиции	179
Манипулятор транспозиции	180
Подчищаем поверхность модели	185
Уточняем формы	187

Глава 4. Подобъекты, Z-сферы

и инструмент ZSketch.....	189
Добавление подобъектов	190
Создаем глаза	191
Подвергаем подобъекты скульптурной деформации	199
Моделируем глаз	200
Плагин SubTool Master	201
Создаем веки методом экстракции сетки.....	208
Объединяем подобъекты	213
Смешиваем геометрию объединенных поверхностей	216
Создаем зубы	220
Параметрические примитивы	220
Применяем деформации.....	226
Добавляем зубы в модель дракона.....	227
Добавляем в модель рога и уши	230
Z-сферы	232
Основы работы с Z-сферами.....	232
Скиннинг	235
Адаптивный скиннинг	236
Моделируем дракона с помощью Z-сфер	242
Работаем с адаптивной оболочкой.....	251
Используем инструмент ZSketch.....	254
Создаем Z-скетч	254
Равномерная оболочка в режиме предпросмотра	261

Добавляем детали.....	266
Привязываем Z-скетч к скелету и позиционируем его	267
Создаем равномерную оболочку.....	272
Работаем с оболочкой.....	274

Глава 5. ShadowBox и режущие кисти..... 276

ShadowBox	277
Создаем ShadowBox	278
Разрешение ShadowBox.....	281
Работаем с эталонными изображениями	285
Создаем корпус автомобиля в ShadowBox.....	290
Работаем с кистью MaskCurve.....	293
Работаем с кистью MaskCircle	302
Работаем с верхней проекцией автомобиля	303
Создаем окна кистью MaskLasso	306
Используем радиальную симметрию вместе с ShadowBox	307
Альфа-текстуры и ShadowBox	310
Кисть Match Maker	312
Режущие кисти	317
Основы работы с режущими кистями	317
Работаем с кистью ClipCircle	320
Создаем автомобильное колесо с помощью режущих кистей	322
Опции Brush Radius и PolyGroup	325
Работаем с кистью ClipCurve	328
Совершенствуем формы автомобиля с помощью режущих кистей	333
Срезаем поверхность под углом.....	334
Совершенствуем форму окон с помощью режущих кистей	341
Релаксация сетки	342
Создаем внутреннее пространство под салон автомобиля.....	344

Глава 6. Операция ReMesh и проецирование	348
Перерасчет топологии (ReMesh).....	349
Перерасчитываем сетку простого объекта	349
Перерасчитываем сетку совокупности подобъектов.....	353
Проецирование	360
Моделируем фары	360
Создаем единую сетку для фары и проецируем на нее детали	364
Манекены	368
Позиционирование с использованием манекенов.....	369
Создаем композицию из нескольких персонажей	371
Редактируем манекены	374
Моделируем крылья.....	378
Получаем сетку операцией ReMesh.....	380
Проецируем детали с манекена на сетку	385
Работаем над формами модели дракона.....	393
Глава 7. Продвинутые техники работы с кистями	395
Настройка кистей	396
Создаем новую кисть.....	402
LazyMouse.....	402
Сохраняем пользовательскую кисть.....	406
Ползунок Brush Modifier.....	408
Настройки силы нажатия на графический планшет (Tablet Pressure).....	409
Создаем иконку для пользовательской кисти	414
Используем кисть detailBrush	416
Альфа-текстуры.....	416
Создаем альфу в Photoshop	418
Создаем альфу в ZBrush.....	422
Применяем к альфе функцию Roll.....	427
Редактируем альфу	429

Опция Tilt (Наклон)	433
Опция Spin (Поворот)	434
Эффекты кистей	438
Эффект Trails (След)	438
Эффект Elasticity (Эластичность)	440
Эффект Noise (Шум)	441
Кисти для создания деталей с твердой поверхностью	443
Планарные кисти	444
Опция BackTrack	448
Plane (Плоскость)	448
Line (Линия)	449
Spline (Сплайн)	450
Path (Путь)	452
Глубина воздействия кисти	453
Маскирование глубины воздействия кисти	457
Палитра Picker	460
Комбинируем настройки	462
Трафареты	463
Создаем трафарет	464
Управление трафаретом	465
Палитра Stencil (Трафарет)	467

Глава 8. Полирисование и инструмент

Spotlight	471
Полирисование	472
Основы полирисования	473
Вторичный цвет	476
Смешивание цветов	479
Размытие штриха	481
Заливка объекта цветом	482
Использование текстур	483
Полирисование и уровни подразделения объекта	485
Техники полирисования	486
Создаем базовый слой цвета	488

Создаем цветовые зоны	491
Создаем крапчатость	495
Рисуем подповерхностные детали	499
Маскирование впадин	500
Маскирование Ambient Occlusion	503
Рисуем детали поверхности	506
Редактирование изображений в Spotlight	509
Запуск и завершение работы Spotlight	510
Запуск Spotlight из библиотеки текстур.....	510
Запуск Spotlight из окна Light Box.....	512
Сохранение и загрузка сессий Spotlight	514
Преобразование изображений.....	516
Функции Spotlight.....	519
Проецирование текстур	524
Проецируем рисунок.....	525
Привязка проекции к кисти.....	528
Комбинирование изображений с помощью опции Fade	529
Spotlight и «цифровая лепка».....	530
Симметричное проецирование	534
Текстура Spotlight как эталонное изображение	537
Глава 9. Визуализация, свет и материалы.....	540
Основы визуализации.....	541
Выбор режима визуализации	544
Визуализация участка холста	547
Настройка визуализации.....	548
Визуализация в режиме BPR.....	552
Начало рендеринга	552
Сглаживание краев изображения	553
Прозрачность подобъектов	553
Настройки прозрачности.....	555
Направление нормалей и прозрачность.....	556
Цвет и прозрачность	558
Преломляющие свойства подобъектов	560
Тени в режиме BPR.....	564

Ambient Occlusion	566
Визуализация в режиме Best	569
Сглаживание (Antialiasing)	569
Туман (Fog).....	570
Экспорт визуализированного изображения	572
Размер документа и цвет фона.....	573
Субпалитра Adjustments (Установки)	577
Каналы визуализации.....	577
Источники света в ZBrush	581
Изменяем положение источника света	581
Выбор и создание нового источника света	583
Настройки освещенности.....	584
Типы источников света	585
Тип Sun (Солнечный свет)	586
Тип Point (Точечный свет)	589
Тип Spot (Прожектор)	589
Тип Glow (Свечение)	591
Тени в режиме Best	592
Понятие о материалах	596
Стандартные материалы и их шейдеры	597
Копирование и вставка шейдера из буфера обмена	600
Копирование и вставка материала из буфера обмена	600
Сохранение материала	601
Создаем пользовательский стандартный шейдер	601
Редактируем рассеивающие свойства материала.....	602
Редактируем зеркальные свойства материала	604
Редактируем отражающие свойства материала	605
Дополнительные настройки материалов	607
Материалы группы MatCap	609
Инструмент захвата материалов	610
Редактируем материалы MatCap.....	613
Субпалитра Shader Mixer (Смешивание шейдеров).....	618
Смешиваем шейдеры	619
Режимы смешивания шейдеров	619

Визуализация подповерхностного рассеивания	622
Визуализируем материал с подповерхностным рассеиванием	622
Материал Fibers (Волокна)	625
Визуализация шерсти и волос	625
Визуализируем бороду	626
Глава 10. Целевые формы, слои и временная шкала ZBrush	630
Целевые формы (морфы)	631
Сохранение целевой формы	631
Переключение между состояниями модели	633
Удаление целевой формы	635
Кисть Morph	635
3D-слои	636
Субпалитра Layers (Слои)	637
Режим записи слоя	638
Интенсивность слоя	638
Обжиг слоя	641
Слои и полирисование	645
Создание слоя для полирисования	646
Создание текстурных карт	651
Временная шкала ZBrush	654
Палитра Movie (Клип)	655
Временная шкала	657
Анимация слоев	659
Запись и экспорт роликов	662
Контактные точки	664
Бонусная глава 1. GoZ	668
Ретопология	669
Подготовка модели к ретопологии	669
Ретопология модели	673
Предпросмотр новой сетки	676
Ретопология множественных подобъектов	678

Создание адаптивной оболочки	678
Проецирование	681
Моделирование при помощи GoZ	683
Отсылаем модель из ZBrush в Maya.....	684
Отсылаем сетку из Maya в ZBrush.....	687
Редактируем топологию в Maya.....	690
Создание текстурных карт в ZBrush	693
Текстурные UV-координаты	694
Создаем цветовую текстурную карту.....	695
Создание карт нормалей и смещения	698
Пересылаем текстурные карты в Maya с помощью GoZ	702
Бонусная глава 2. Z-скрипты и Z-плагины	705
Z-скрипты	706
Работа с Quick Sketch	706
Запись Z-скрипта	708
Загрузка и применение Z-скрипта.....	711
Плагин Projection Master	712
Раскрашиваем поверхность объекта при помощи Projection Master	713
Деформируем поверхность объекта при помощи Projection Master	718
Другие Z-плагины	720
Установка Z-плагинов.....	720
Плагин UV Master	723
Опция Control Painting	727
Плагин PaintStop	731
Плагин 3D Print Exporter.....	736
Плагин Decimation Master	737
Плагин ZAppLink3	743
Плагин Multi Map Exporter	746
Плагин Transpose Master	750
Плагин Adjust Draw Size	752
Плагин SubTool Master	753

Приложение. DVD к книге	754
Что вы найдете на DVD	754
Файлы глав книги	754
Системные требования.....	754
Работа с диском	756
Поиск и устранение неисправностей	756
Художественная галерея ZBrush	758



Глава

1 Основы компьютерной графики

Как известно, в основе любого творчества лежит взаимодействие различных материалов (красок, глины и т. д.), которое в конечном итоге и определяет облик произведения искусства. Если вы захотите стать художником или скульптором, то вам придется немало времени уделить принципам работы с тем или иным материалом. То же самое можно сказать и про компьютерную графику. Эта глава посвящена обзору фундаментальных основ цифрового искусства. Точно так же, как настоящий художник учится смешивать пигменты и масла и постигает взаимодействие краски с холстом, цифровой художник учится работать с цветом, каналами, форматами файлов и другими элементами, которые влияют на качество изображения.

Эта глава включает в себя следующие темы:

- введение в ZBrush;
- понятие о цифровом изображении;
- понятие о трехмерном пространстве;
- как стать цифровым художником.

Введение в ZBrush

Представьте себе полную инструментов студию художника или скульптора – краски и кисти, холсты и глина, осветительные приборы и камеры, арматура для макетов и бесконечное множество различных скребков и ножей. Причем все это расставлено и разложено самым удобным для работы образом. Именно такой студией является программа ZBrush. С ее помощью вы можете создавать картины, скульптуры и даже комбинировать одно с другим. Более того – вы не ограничены возможностями только лишь этой программы. Трехмерные модели и двумерные текстуры можно легко импортировать из других приложений и использовать в ZBrush в качестве самостоятельных инструментов. ZBrush может работать и как независимая программа, и как одна из составляющих рабочего процесса, в ходе которого задействует сразу несколько приложений, например при работе над анимацией.

Чаще всего ZBrush используется для создания и правки объектов, которые затем анимируются и визуализируются в других пакетах трехмерной графики – Maya и 3ds Max компании Autodesk, Cinema 4D компании Maxon, Modo компании Luxology и др. Уникальность ZBrush заключается в том, что эта программа способна работать с моделями, состоящими из миллионов полигонов. Подобный уровень плотности, необходимый для создания различных органических поверхностей, просто невозможно обработать с помощью других 3D-программ. Морщины, складки кожи, поры, вмятины, чешуя, шрамы и царапины – все это можно легко создать в ZBrush, а затем экспортировать либо как часть геометрии, либо в виде карт неровностей и смещения для последующей визуализации в других приложениях. В результате достигается удивительный уровень детализации и реализма виртуального объекта (рис. 1.1). Цветовые текстурные карты можно интуитивным образом нарисовать прямо в ZBrush, а потом экспортировать для шейдинга объектов в других программах. Профессиональные художественные студии – ILM, Gentle Giant, Weta и Sony Imageworks – использовали ZBrush при работе над такими фильмами, как, например, «Властелин колец», «Пираты Карибского моря», «Небесный капитан и мир будущего».

В последние годы для ZBrush нашлось применение и в других областях, не связанных напрямую с анимацией и визуальными эффектами: в производстве игрушек, разработке игровых персонажей и виртуального окружения, научной визуализации, ювелирном де-

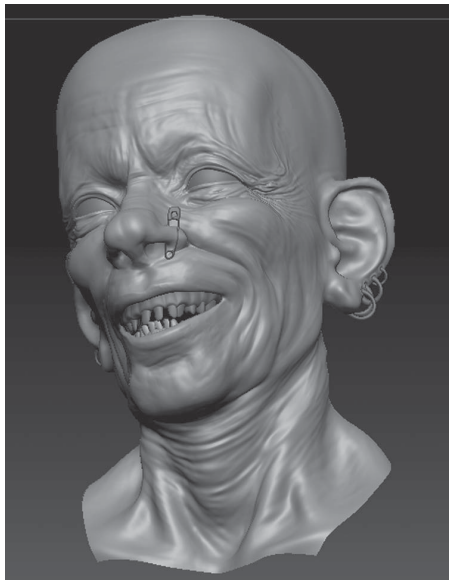


Рис. 1.1. Высокодетализированная модель ZBrush

ле, концептуальном дизайне и даже в создании реальных физических скульптур. Виртуальная модель, созданная в среде ZBrush, может быть воплощена в физическом объекте с помощью технологии 3D-прототипирования. 3D-принтеры становятся все более доступными, и потому можно легко представить, как в недалеком будущем ZBrush будет органично включен в процесс производства, став тем его звеном, которое реализуется на рабочем столе компьютера.

Кроме того, ZBrush используется и как иллюстраторская программа, в которой имеется весь набор инструментов для рисования и создания цифровых скульптур, а также собственная технология визуализации. С помощью ZBrush легко получить пользовательские материалы, которые могут быть разработаны с помощью подручных средств или «захвачены» прямо из изображений. Все эти материалы можно применить к виртуальным объектам, и при визуализации они будут адекватно взаимодействовать с источниками света. Многие художники уже успели оценить всю гибкость и мощь инструментов ZBrush при создании удивительных композиций – как непосредственно в самой программе, так и с помощью других 2D-приложений, например Photoshop или Corel Painter, с которыми ZBrush прекрасно

комбинируется. И цифровые трехмерные модели, и двумерные изображения можно свободно экспортировать и импортировать из этих программ, так что перед художником, использующим ZBrush, открывается мир поистине безграничных возможностей.

Понятие о цифровом изображении

Теперь давайте кратко разберем, как компьютер создает графику и отображает ее на дисплее. Цифровые изображения состоят из маленьких цветных квадратиков, называемых пикселями. В данном разделе будут рассмотрены базовые принципы работы с пикселями и связанные с этим темы.

Анатомия пиксела

Пиксел – это цветной квадратик, который занимает на экране определенное положение. Изображение, которое состоит из тысяч и тысяч таких квадратиков, называется *растровым*. Таким образом, каждый пиксел несет в себе информацию о собственном цвете и положении, которая хранится в памяти компьютера. Если вы загрузите растровую иллюстрацию в программу просмотра цифровой графики и сильно увеличите масштаб, то сможете увидеть пиксели, из которых состоит эта иллюстрация (рис. 1.2).

Местоположение пиксела описывается координатами по горизонтальной оси X и вертикальной оси Y. Вам это может показаться очевидным, однако имейте в виду, что всякий раз, когда вы изменяете масштаб или прокручиваете изображение, позиция и размер пикселей меняются относительно экрана. И вместе с тем компьютер помнит «настоящие» координаты и размер пикселей относительно самого изображения. Но, впрочем, не уделяйте этому много внимания: компьютер сам просчитывает и хранит подобную информацию.

Сглаживание краев изображения

В некоторых случаях изогнутые линии кажутся на экране зазубренными – по причине того, что состоят из крошечных квадратиков-пикселей. Для устранения этой проблемы графические приложения используют технологию *сглаживания* (*anti-aliasing*), которое осу-

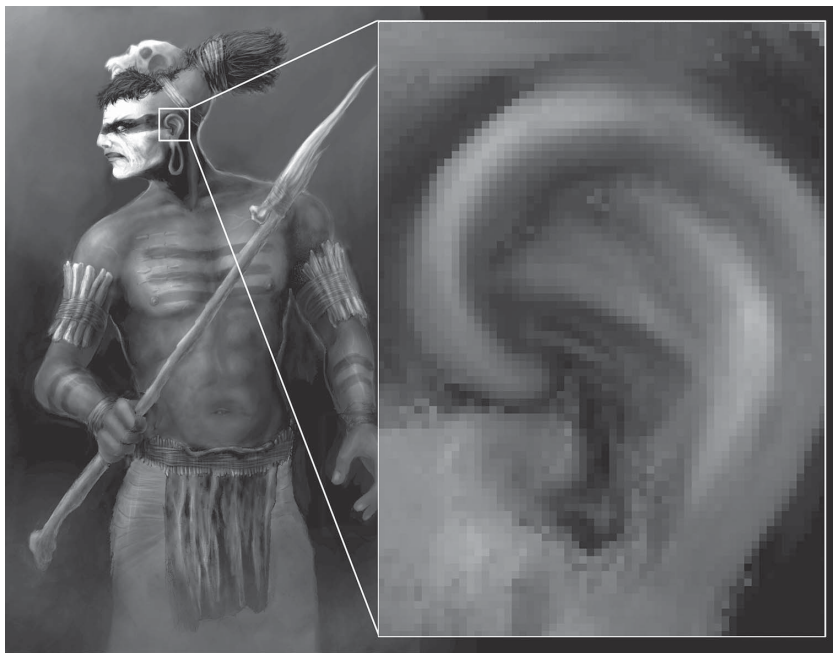


Рис. 1.2. Рисунок, созданный в Painter. На масштабированной области видны квадратички-пиксели, из которых состоит изображение

ществляется за счет смешивания пикселей по краю объекта с другими пикселями того же оттенка, но различных степеней яркости и прозрачности. В результате глаз воспринимает такой объект как сглаженный.

На рис. 1.3 края букв в слове «jagged» кажутся зазубренными, поскольку видны пиксели, из которых состоит изображение. Напротив, в слове «smooth» края кажутся гладкими, потому что здесь применена описанная технология сглаживания.

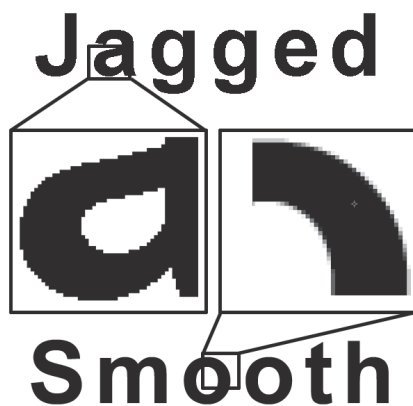


Рис. 1.3. Буквы в слове «jagged» зазубрены, в слове «smooth» – сглажены

Каналы и глубина цвета

Помимо информации о своем местоположении, пикселы хранят информацию и о цвете. Компьютер образует цвета путем смешивания трех оттенков – красного, зеленого и синего. Если пиксел содержит в себе 100% красного цвета и по 0% синего и зеленого, то на экране он будет выглядеть красным. Если же в пикселе содержится, например, 50% красного, 50% синего и 0% зеленого цветов, то он будет выглядеть фиолетовым. Если значения всех трех цветов равны 0%, то пиксел будет черным, а если все три содержатся в количестве 100% – пиксел будет белым.

Глубина цвета отвечает за то, какое количество информации о цвете пикселов хранится в изображении. Режим **Grayscale** (Оттенки серого), например, позволяет сохранить информацию лишь о белом и черном цветах и всех оттенках серого, лежащих между ними; как правило, используется 256 градаций серого. В результате получаются черно-белые изображения, как те, которые приведены в этой главе. Поскольку информация о цвете ограничивается лишь этими 256 оттенками, общий размер файла получается меньше, чем в случае цветных изображений.

Если вы когда-нибудь учились рисованию, то знаете, что первичными цветами являются красный, синий и желтый, а зеленый цвет является вторичным, поскольку получается путем смешивания синего и желтого цветов. Это действительно так, когда речь идет о бумаге. Однако экран компьютера использует совсем другую технологию – не наложения цветов, а их подсветки – и потому первичными являются красный, зеленый и синий цвета (red, green, blue – RGB), а желтый цвет получается путем смешивания красного и зеленого.

Таким образом, изображения в формате RGB хранят информацию о количестве красного, зеленого и синего цветов. Эта информация разделена на три соответствующих *канала* – красный, зеленый и синий. Каждый канал хранит значения, выражаемые в процентах, о количестве соответствующего цвета в каждом пикселе изображения.

Для того чтобы увидеть, как это работает, выполните следующее упражнение:

1. Начните сеанс ZBrush.
2. Щелкните по меню **Color** в верхней части интерфейса, чтобы развернуть соответствующее меню-палитру.
3. Поводите курсором над областью цветовой палитры (рис. 1.4).
4. Понаблюдайте, как меняются числовые значения в полях R, G и B, расположенных ниже цветовой палитры. Эти значения от-

ражают количество цвета, необходимого для создания данного оттенка. Обратите внимание: наибольшим возможным значением является 255, а наименьшим – 0 (рис. 1.4).

- Щелкая по ползункам R, G и B, введите следующие значения: R = 255, G = 0 и B = 255. В результате получится ярко-розовый цвет – фуксия.

Изображения в формате RGBA содержат в себе дополнительный канал альфа, то есть канал прозрачности. Альфа-канал хранит информацию о степени прозрачности пикселей и тем самым позволяет отдельным областям изображения выглядеть прозрачными. Левая часть рис. 1.5 демонстрирует изображение, полученное при визуализации в программе 3D-графики; сферы, как мы видим, являются прозрачными. Правая часть рисунка показывает, как выглядит альфа-канал изображения. Белые области являются полностью непрозрачными, чер-



Рис. 1.4. Числа в полях R (red, красный), G (green, зеленый) и B (blue, синий) показывают значения соответствующих каналов

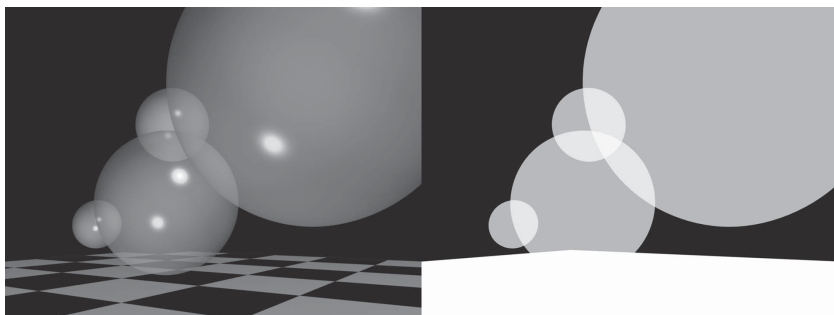


Рис. 1.5. Левая часть рисунка – это изображение в формате RGB, правая – альфа-канал изображения

ные – наоборот, полностью прозрачными. Серые области определяют степени прозрачности.

Глубина цвета отвечает за то, какое количество информации хранится в каждом из этих каналов. Для хранения информации компьютеры используют двоичную кодировку, или биты, представляющие собой набор нулей и единиц. 24-битовое изображение в формате RGB означает, что для каждого канала используется 8 битов информации ($3 \times 8 = 24$). Каждый такой 8-битовый канал может описать 256 оттенков цвета, что позволяет создавать изображение, состоящее более чем из 16 миллионов цветов. 32-битовое изображение в формате RGBA содержит в себе дополнительный 8-битовый альфа-канал.

Чем больше битов используется для описания информации, тем больше информации вы можете сохранить и тем более насыщенной является цветовая гамма, в которой изображение будет представлено на экране компьютера. Соответственно, чем больше битов, тем больше памяти требуется для хранения и работы с изображением. Существуют изображения, которые содержат по 16 битов на каждый канал (суммарно 48 битов в формате RGB и 64 бита в формате RGBA), и их иногда ошибочно называют 16-битовыми изображениями (как и сохраненные в 16-битовом формате TIFF или 16-битовом формате SGI).

Осторожно! Изображения, использующие по 16 битов на канал, – это не то же самое, что 16-битовые изображения, в которых на каждый канал приходится примерно по 5 битов информации. Добро пожаловать в запутанный мир компьютерной терминологии! Хотя компьютеры действуют в соответствии со строгой логикой, такое далеко не всегда можно сказать о людях, которые эти компьютеры создают. Если вы – художник, занятый на телевидении или в производстве фильмов, вы будете гораздо чаще работать с изображениями, содержащими по 16 битов на канал, чем с 16-битовыми (содержащими по 5 битов на канал) изображениями.

Форматы файлов изображений

Цифровые изображения можно сохранить в виде файлов различных *форматов*. Формат – это способ организации информации в файле. Наиболее часто используются форматы Tagged Image File Format (TIFF), Joint Photographic Experts Group (JPEG) и Graphics Interchange Format (GIF).

Многие приложения хранят информацию в своем собственном формате. Например, Photoshop читает большинство графических форматов, но для хранения специальной информации имеет свой собственный – Photoshop Document format (PSD). Аналогичным об-

разом Corel Painter специфическую информацию хранит в собственном формате Resource Interchange File Format (RIFF). Рабочий формат ZBrush называется ZBR.

Формат изображения может использовать сжатие данных, чтобы уменьшить размер файла, например форматы JPEG и GIF. Некоторые форматы не сжимают информацию, например Silicon Graphics Image (SGI). Сжатие иногда приводит к потере качества изображения, что часто происходит, например, при хранении файлов в формате JPEG. Если приглядеться к JPEG-изображениям, представленным на сайтах Интернета, то часто можно увидеть их размытость или зернистость, а иногда и некорректное отображение цветов. В данном случае создатели сайтов жертвуют качеством изображения ради повышения скорости загрузки веб-страниц.

В том случае если сжатие данных сопровождается ухудшением качества изображения, говорят, что такой формат использует сжатие с потерями. Существуют и форматы, в которых сжатие не приводит к ухудшению качества, – в этом случае говорят о сжатии без потерь. К подобным форматам относится, например, Portable Network Graphics (PNG). Однако размер файла, сохраненного в таких форматах, будет больше, чем при сжатии с потерей качества изображения.

Сжатие данных используется также и для создания последовательности кадров при записи видео.

На левой части рис. 1.6 представлено изображение без сжатия данных, а на правой – со сжатием данных. Если внимательно присмотр-



Рис. 1.6. Левое изображение представлено в формате без сжатия данных, правое – со сжатием

реться, то на правой фотографии можно разглядеть искажения-артефакты, особенно в той части, где изображены шерсть белки и опоры забора.

Понимание форматов графических файлов важно для работы с компьютерной графикой не только в среде ZBrush, но и в других приложениях. Если вы используете изображение со сжатием данных как источник текстуры для виртуального объекта или как инструмент для создания цифровой скульптуры, то итоговый результат может получиться неудовлетворительным по причине низкого качества исходного изображения. Вдобавок ко всему некоторые 3D-программы и системы визуализации предпочитают работать с определенными форматами, что следует учитывать при экспорте изображений из ZBrush. Но об этих аспектах работы мы поговорим немного позже.

Векторные изображения

Существует и другой способ построения изображения, а именно *векторный*. Векторная графика создается на основе формул и математических вычислений, производимых компьютером и графической программой. Линии на векторных изображениях всегда гладкие. Векторные изображения всякий раз перерасчитываются и вырисовываются заново, когда вы масштабируете, двигаете или вращаете картинку, так что качество изображения не зависит ни от его размера, ни от местоположения.

Популярными программами для работы с векторной графикой являются Adobe Illustrator и Adobe Flash. Векторный способ построения изображений используется и при моделировании поверхности в 3D-приложениях, таких как Maya или 3ds Max, и эти программы также обладают возможностью визуализировать финальное изображение в векторном формате. Но поскольку в ZBrush векторная графика не используется, на этом мы и закончим разговор о ней.

Что такое «разрешение изображения»

Трудно преувеличить значение разрешения, если вы работаете в ZBrush. К сожалению, понятие о разрешении – довольно непростая вещь. Существует множество путанных терминов, описывающих из-

мерение и вычисление степеней разрешения, равно как и различных типов разрешения. Но к этой теме я буду возвращаться на протяжении всей книги, поэтому не огорчайтесь, если вы чего-то не поняли сразу.

Если говорить просто, то *разрешение* – это количество информации на единицу площади. Разрешение цифровых изображений чаще всего измеряется в количестве пикселей, которые занимают определенную единицу поверхности изображения. Но измерить разрешение можно также и в числе полигонов или вершин на определенном участке поверхности трехмерного объекта. Существует еще и разрешение экрана, определяющее, как отображается графический объект в определенном разрешении. Кроме того, если вы накладываете двумерную текстуру на трехмерный объект, то должны учитывать, что разрешение текстуры измеряется в пикселях, а разрешение объекта – в полигонах, иначе результат может вас разочаровать. С этим вам придется часто сталкиваться при работе в ZBrush, поэтому о разрешении нужно помнить всегда.

Разрешение экрана

Разговор о видах разрешения давайте начнем с экрана. Компьютер, на котором вы работаете, вне всякого сомнения, имеет монитор (если нет, то ваша карьера в качестве цифрового художника может закончиться самым драматическим образом). Монитор отображает текст и графику. Разрешение монитора представлено определенным количеством квадратиков-пикселей и измеряется в горизонтальном и вертикальном направлениях. Физический размер монитора обычно измеряется по диагонали экрана. Если размер монитора – 22 дюйма, то это значит, что именно на таком расстоянии располагаются друг от друга два противоположных по диагонали угла экрана.

Монитор, с которым вы работаете, должен уметь отображать графику и текст в различных разрешениях. Текущее разрешение экрана определяется установками контрольной панели операционной системы и описывает количество пикселей по горизонтали и вертикали. Например, один из наиболее распространенных видов разрешения – 640×480, которым часто пользовались на заре компьютерной эры, когда мониторы были маленького размера; разрешение 720×486 используется в США для обычного телевидения, а 1920×1080 – для телевидения с высоким разрешением (HDTV).

Разрешение монитора определяет отображение ZBrush на экране вашего компьютера. Если сделать разрешение маленьким, то на экране останется меньше места как для интерфейса ZBrush, так и для документов. По этой причине компьютерные художники не жалеют средств на приобретение больших мониторов, а иногда даже присоединяют два монитора к одному компьютеру.

Разрешение документа

Теперь давайте разберем, что такое разрешение документа. Ранее я упомянул, что если сильно увеличить масштаб растрового изображения, то можно увидеть отдельные пиксели, из которых это изображение состоит. На самом деле, изменяя масштаб, вы не изменяете размера самих пикселей и не меняете настроек монитора. Тем не менее графические программы позволяют вам увидеть, как будет выглядеть изображение при увеличении его размеров по отношению к оригинальному размеру документа.

Если разрешение вашего документа – 320×240, а вы увеличите масштаб до 200%, то разрешение документа станет 640×480. При этом каждый пиксел будет использован четырежды, что придаст изображению зернистый вид. Аналогичным образом, если вы уменьшите масштаб, отображаться будет лишь половина исходных пикселей. Способность изменять масштаб изображения – полезная особенность графических программ, позволяющая работать с мелкими деталями. Но тут есть одна хитрость: разрешение документа, измененное при помощи масштабирования, не совпадает с текущим разрешением экрана, о чем вы всегда должны помнить при работе с цифровыми изображениями.

Как правило, для описания разрешения документа используется величина dots per inch (dpi) – количество точек на дюйм. Иногда эта же величина обозначается как pixels per inch (ppi), количество пикселей на дюйм. Такая размерность используется даже в странах с метрической системой измерения, например во Франции. Изображение, которое отображается на экране в своем 100% размере, обычно имеет разрешение 72 dpi. Изображение, которое в последующем будет распечатано на бумаге, должно иметь большее разрешение – как минимум 300 dpi. В коммерческой печати обычно используется разрешение от 600 до 1200 dpi.

Разрешение изображения

В среде 3D-художников популярен термин *2К-текстура*, означающий, что изображение имеет разрешение 2048×2048 пикселей. В то время как для всех нормальных людей 2К означает 2000, для художников, имеющих дело с компьютерной графикой, 2К равно 2048. Дело в том, что большинство текстур имеют разрешение, равное степеням двойки, и, следовательно, 1К = 1024 (2^{10}), 4К = 4096 (2^{12}), а 512 (2^9) означает 512×512.

Любой 3D-художник скажет вам, что текстура размера 2К всегда квадратная. А теперь представьте себе, что вы попали на киностудию, и вас попросили сделать анимацию размером 2К. Если кадры вашей анимации будут иметь разрешение 2048×2048 пикселей, вас очень быстро выставят за дверь. Почему? Потому, что для людей, занятых в кинопроизводстве, 2К является сокращением от «2К Academy» – так называется стандартизированный размер кадра, равный 2048×1556 пикселей. Опять мы видим, что значение термина определяется контекстом. Поскольку эта книга посвящена ZBrush, то я буду пользоваться языком 3D-художников, а значит, 2К в моем случае означает 2048×2048. Но если (и когда) вы будете работать над созданием анимаций в программах, подобных Maya, не забывайте, что для разных людей 2К может иметь разные значения. Во избежание недоразумений лучше сразу уточнять, какое именно разрешение подразумевают другие под этим сокращением. Жаргонизмы в работе больше мешают, чем помогают.



Некоторые компьютерные специалисты под буквой К понимают «килобайт». В этих единицах измеряется фактический размер файла, сохраненного на диске. Так что вот и еще один повод для путаницы.

Соотношение размеров

Формат изображения можно также указать в виде соотношения горизонтального и вертикального размеров. Например, изображение размерами 320×240 или 640×480 пикселей обладает соотношением размеров 4:3. Если это соотношение равно 16:9 или 1,85:1, то это значит, что изображение является широкоэкранным. Обычным разрешением для изображения 16:9 будет являться 1280×720. Но об этом следует помнить скорее при визуализации анимации в таких пакетах, как Maya. Что касается ZBrush, то формат изображения будет иметь

значение лишь при создании композиций, которые в последующем будут использованы для анимации или других целей.

Полигональное разрешение

Наконец, понятием «разрешение» можно описать количество точек или полигонов, составляющих трехмерный объект. О полигонах еще будет рассказано далее в этой главе, а пока уясните себе, что поверхность трехмерного объекта состоит из отдельных участков, имеющих три или более вершин (в ZBrush полигоны ограничены тремя или четырьмя вершинами, но в других приложениях количество вершин в полигоне может быть и больше). Количество полигонов в объекте можно увеличивать, в результате чего поверхность сглаживается, а вы приобретаете возможность добавлять более мелкие детали геометрии.

Модель в ZBrush может состоять из миллионов и миллионов полигонов, как показано на рис. 1.7. Тем не менее благодаря уникальной

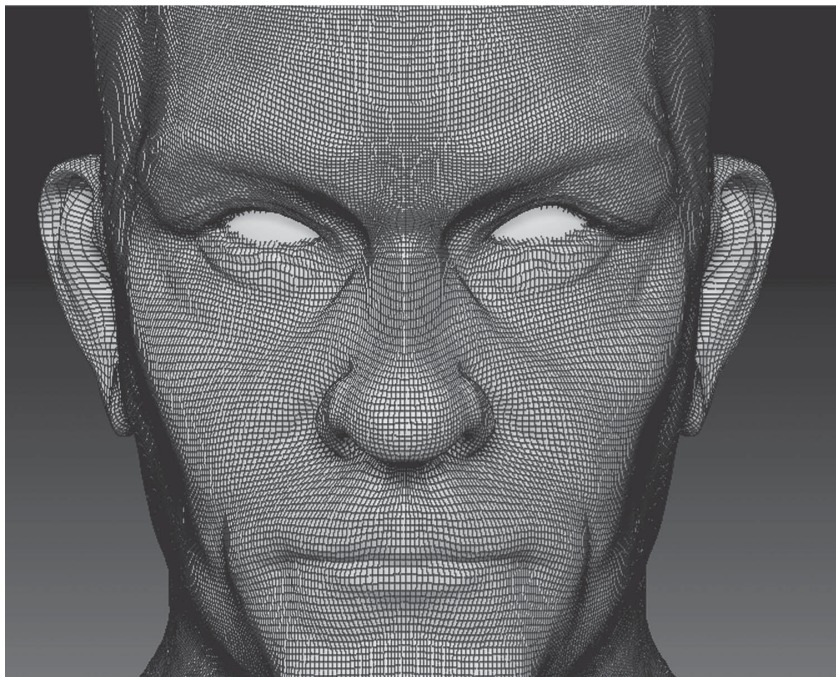


Рис. 1.7. Модель ZBrush в высоком разрешении. Линии на поверхности – это границы полигонов

технологии ZBrush вы можете легко редактировать такие модели. При этом производительность программы существенно не страдает, чего нельзя сказать о работе с другими 3D-пакетами. Кроме того, ZBrush позволяет хранить множество уровней полигональности для одного и того же объекта в одном файле, переключаться между этими уровнями в процессе работы, а также экспортировать объект на разных уровнях полигональности для использования в других приложениях.

На этом мы закончим вводную часть, посвященную разрешению. Но будьте уверены, что к этой теме мы будем возвращаться снова и снова на протяжении всей книги.

Понятие о трехмерном пространстве

В типичных 3D-приложениях, например Maya, трехмерное пространство описывается тремя координатами – X, Y и Z. При этом X-координата обычно обозначает горизонтальное направление, Y-координата – вертикальное, а Z-координата – глубину (в некоторых приложениях координаты Y и Z меняются местами). В Maya для ориентации в виртуальном мире используется сетка координат. Очень важно понять, что определенная точка в трехмерном пространстве – например, вершина объекта – имеет свое абсолютное положение в трехмерном мире и привязана к определенным глобальным мировым координатам. Кроме того, эта же точка имеет и локальные координаты, которыми определяется ее местоположение относительно других частей объекта.

Приведу такой пример: на вас надета высокая остроконечная шляпа. Согласно глобальным мировым координатам, а именно Y-координате, кончик этой шляпы является самой верхней точкой объекта, то есть вас. В то же время локальная координата Y также указывает на то, что кончик шляпы является вершиной самой шляпы. Теперь представьте себе, что вы висите вверх тормашками. Теперь, согласно глобальным мировым координатам, кончик шляпы является самой низкой точкой объекта. Однако в локальной системе координат кончик шляпы по-прежнему является вершиной шляпы. Локальная система координат позволяет нам понять, что кончик шляпы является вершиной объекта независимо от того, как вы вращаете этот объект в трехмерном пространстве. Именно на локальной системе коорди-

нат основываются наши представления о предметах и их назначении в этом мире. 3D-программы, помимо собственно моделирования объектов, позволяют отследить их перемещение в трехмерном пространстве с помощью указанных двух систем координат (рис. 1.8).

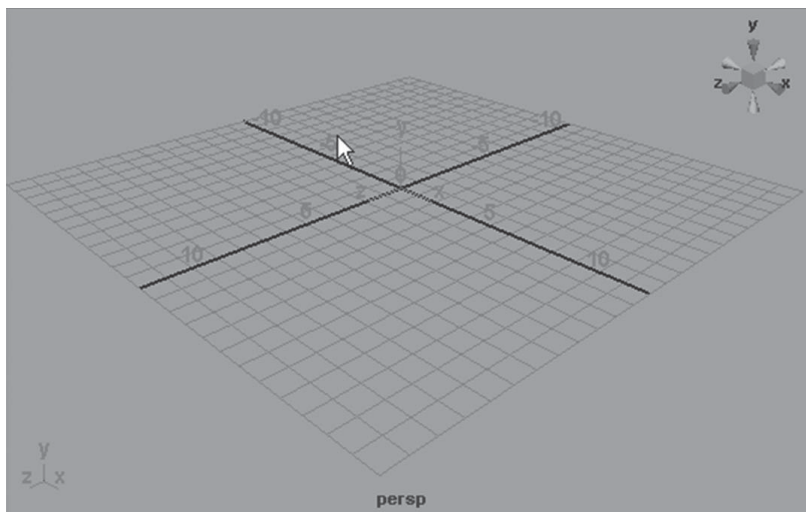


Рис. 1.8. Типичное 3D-пространство. Сетка и 3D-компас помогают художнику ориентироваться в виртуальном мире

Анатомия полигона

На самом деле такой вещи, как трехмерный объект, не существует, если вы, конечно, не работаете с 3D-принтерами, которые могут создать реальный физический объект по вашей виртуальной модели. В действительности же вы всегда работаете лишь с двумерным экранном отображением трехмерного объекта. Последующие издания этой книги, несомненно, будут описывать работу с 3D-принтерами, как только технология 3D-прототипирования станет более дешевой и доступной для художников. Однако пока что вы, скорее всего, будете работать только с той проекцией трехмерного объекта, которую видите на экране своего компьютера. И именно эту проекцию мы и будем иметь в виду, говоря о трехмерном виртуальном объекте.

Типичные графические программы, например Photoshop, хранят информацию о горизонтальном и вертикальном положении пикселей

по соответствующим осям координат (X и Y), а типичные 3D-программы – еще и информацию о дополнительной оси глубины (Z), что создает иллюзию объема. Виртуальный объект в трехмерном мире состоит из *полигонов*. Полигоны формируют поверхность объекта, и именно за счет них эту поверхность можно деформировать и анимировать.

Полигон представляет собой геометрическую форму, определяемую тремя или более точками (точки называются вершинами полигона). Пример полигонов приведен на рис. 1.9.

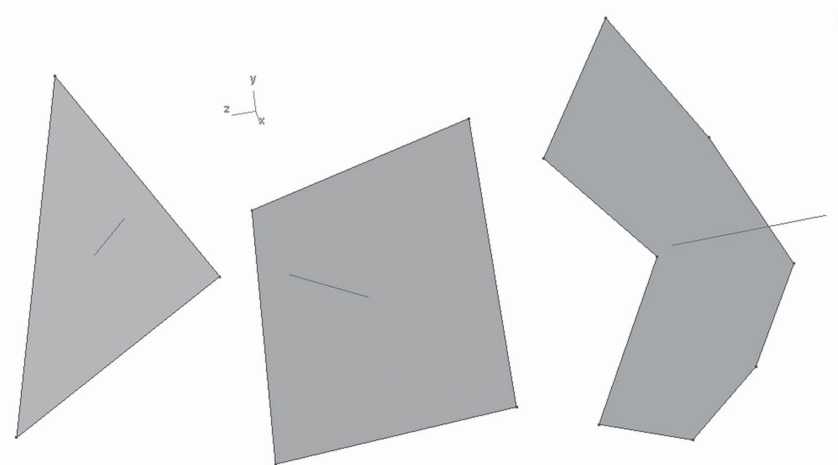


Рис. 1.9. Примеры трех-, четырех- и многоугольного полигонов в Maya



ZBrush ограничивает количество вершин в своих полигонах тремя или четырьмя, но в других приложениях полигоны могут иметь и больше вершин. Об этом следует помнить, когда вы импортируете объект из другого 3D-приложения в ZBrush. При импорте ZBrush автоматически преобразует многоугольные полигоны (то есть имеющие больше 4 вершин) в четырехугольные.

В других 3D-пакетах вам могут встретиться и альтернативные способы построения 3D-геометрии, например NURBS или поверхность с иерархическим разбиением (subdivision surface). При визуализации эти формы преобразуются в треугольные полигоны – стандартный способ построения геометрии в 3D-приложениях. Что касается ZBrush, то эта программа работает исключительно с полигональными объектами.