



В таблице 1 представлена сравнительная характеристика основных облачных инструментов. Для разработки крупных проектов лучше всего подойдут те, что предоставляют функции облачной среды разработки, такие как: интеграция с системами контроля версий, использование систем сборки и изолированных окружений [7]. Изолированные окружения позволяют использовать для компиляции окружения с различными операционными системами, наборами библиотеки и компиляторов. Они могут упростить вовлечение новых разработчиков в проект, благодаря снижению затрат времени на развёртывание рабочего окружения [8].

Таблица 1 – Сравнение возможностей облачных инструментов разработки

Название	Тип	Поддержка VCS	Язык программирования	Изолированное окружение	Возможности редактора	Вывод результатов исполнения
Koding.com	IDE	Да	Мультиязычность	Да	Подсветка, автодополнение	Графический интерфейс
Eclipse Che	IDE	Да	Мультиязычность	Да	Подсветка, автодополнение, синтаксический анализ кода	Графический интерфейс
Cpp.sh	Редактор	Нет	C++	Нет	Подсветка синтаксиса	Консоль
jsFiddle	Редактор	Нет	JS	Нет	Подсветка синтаксиса	Средства визуализации
Cloud9	IDE	Да	Мультиязычность	Да	Подсветка, автодополнение	Средства визуализации
Codeanywhere	IDE	Да	Мультиязычность	Да	Подсветка, автодополнение	Консоль
Python Fiddle	Редактор	Нет	Python	Нет	Подсветка синтаксиса, автодополнение	Консоль
Orionhub.org	IDE	Да	JS	Нет	Подсветка, автодополнение, синтаксический анализ кода	Средства визуализации
Nitrous.io	IDE	Да	Мультиязычность	Да	Подсветка, автодополнение	Средства визуализации
Shiftedit.net	IDE	Да	JS, PHP	Нет	Подсветка, автодополнение	Средства визуализации
Scala JS Fiddle	Редактор	Нет	Scala	Нет	Подсветка, автодополнение	Средства визуализации
try.kotlinlang.org	Редактор	Нет	Kotlin	Нет	Подсветка, автодополнение	Консоль
CEclipse	IDE	Нет	Java	Нет	Подсветка, автодополнение, синтаксический анализ кода	Консоль

Разработка облачного редактора для параллельных приложений с возможностью запуска на кластере

Целью нашей работы является создание облачной IDE для разработки параллельных приложений с последующей возможностью запуска их на кластере «Сергей Королёв». Редактор будет интегрирован в проект Templet Web [9].



Проект призван упростить разработку параллельных приложений при помощи шаблонов Templet, что позволит ускорить изучение параллельных алгоритмов студентами в рамках лабораторных практикумов.

Литература

- Goldman, M. Real-time collaborative coding in a web IDE [Текст] / M. Goldman, G. Little, R.C. Miller // Proceedings of the 24th annual symposium on User interface software and technology. – ACM, 2011. – С.155-164.
- Gadhikar, L.M. et al. Browser based IDE to code in the cloud [Текст] / L.M. Gadhikar et al. //New Paradigms in Internet Computing. – Springer Berlin Heidelberg. – С.59-69.
- Aho, T. et al. Designing IDE as a Service [Текст] / T. Aho et al. //Communications of Cloud Software. – 2011. – Т. 1. – №. 1.
- Wu, L. et al. CEclipse: an online IDE for programing in the cloud [Текст] / L. Wu et al. //Services, 2011 IEEE World Congress on. – IEEE, 2011 – С.45-52.
- Merkel, D. Docker: lightweight linux containers for consistent development and deployment [Текст] / D. Merkel //Linux Journal – 2014 – Т. 2014 – № 239 – С.2.
- Doerane, S. Scala.js: Type-directed interoperability with dynamically typed languages [Текст] / S. Doerane – 2013. – № EPFL-REPORT-190834.
- Mikkonen, T. Elements for a cloud-based development environment: online collaboration, revision control, and continuous integration [Текст] / T. Mikkonen, A. Nieminen // Proceedings of the WISCA/ESCA 2012 Companion Volume. – ACM, 2012 – С.14-20.
- Fahndrich, M. Lessons from a Web-based IDE and Runtime [Текст] / M. Fahndrich // Proceedings of the ACM SIGPLAN 2014 Workshop on Partial Evaluation and Program Manipulation. – ACM, 2014. – С.1-2.
- Артамонов, Ю.С. Применение облачного сервиса Templet Web при проведении лабораторных практикумов на суперкомпьютере «Сергей Королёв» [Текст] / Ю.С. Артамонов, С.В. Востокин // X Международная научно-практическая конференция «Современные информационные технологии и ИТ-образование», МГУ, Москва, 2015. Т. 2. – С.409-414.

Д.В. Щербаков, М.П. Шлеймович

МЕТОДЫ УВЕЛИЧЕНИЯ РАЗРЕШАЮЩЕЙ СПОСОБНОСТИ ИЗОБРАЖЕНИЙ И ИХ ПРИМЕНЕНИЕ

(Казанский национальный исследовательский технический университет им А.Н. Туполева - КАИ)

На сегодняшний день существует множество научных и технологических задач, для решения которых требуются изображения с высоким разрешением. Именно такие изображения используются для анализа и обработки в большин-



стве случаев, так как они позволяют увидеть детали, которые не различимы, или плохо различимы на изображениях с низким разрешением.

В статье ставится задача рассмотреть методы увеличения разрешающей способности изображений и их применение.

В большинстве случаев разработки приложений для увеличения разрешающей способности изображений применяется метод бикубической интерполяции. Бикубическая интерполяция рассматривает массив из 4×4 окружающих пикселей – всего 16. Поскольку они находятся на разных расстояниях от неизвестного пикселя, ближайшие пиксели получают при расчёте больший вес. Чтобы лучше понять принцип бикубической интерполяции, рассматривается принцип кубической интерполяции.

Если значения функции $f(x)$ и ее производной известны в точках $x=0$ и $x=1$, тогда функция может быть интерполирована на интервале $[0, 1]$ используя полином третьего порядка. Формула для вычисления может быть легко получена.

Полином третьего порядка и его производная:

$$f(x) = ax^3 + bx^2 + cx + d; f'(x) = 3ax^2 + 2bx + c \quad (1)$$

Значения полинома и его производной в точках $x=0$ и $x=1$:

$$f(0) = d; f(1) = a + b + c + d; \quad (2)$$

$$f'(0) = c; f'(1) = 3a + 2b + c.$$

Эти четыре тождества могут быть записаны как:

$$a = 2f(0) - 2f(1) + f'(0) + f'(1); \quad (3)$$

$$b = -3f(0) + 3f(1) - 2f'(0) - f'(1);$$

$$c = f'(0); d = f(0)$$

На практике алгоритм используется для интерполяции функции, имея некие известные значения в заданных точках. В этом случае может быть известна производная функции. Тогда можно было бы принять производную в заданных точках, как 0, однако для получения более гладких и правдоподобных графиков функций производная принимается за уклон линии между предыдущей и следующей точкой. Таким образом, для расчетов понадобится четыре точки. Предположим, имеются четыре значения функции в точках p_0, p_1, p_2 и p_3 , расположенных соответственно на $x = -1, x = 0, x = 1$ и $x = 2$. Подставляются полученные значения $f(0), f(1), f(2)$ и $f(3)$:

$$f(1) = p_2; f'(0) = \frac{p_2 - p_0}{2}; f'(1) = \frac{p_3 - p_1}{2} \quad (4)$$

Сопоставив эти данные с полученными ранее формулами (3), имеем:

$$a = -\frac{1}{2}p_0 + \frac{3}{2}p_1 - \frac{3}{2}p_2 + \frac{1}{2}p_3; b = p_0 - \frac{5}{2}p_1 + 2p_2 - \frac{1}{2}p_3; \quad (5)$$

$$c = -\frac{1}{2}p_0 + \frac{1}{2}p_2; d = p_1$$

Результат:

$$f(x, a, b, c, d) = \left(-\frac{1}{2}a + \frac{3}{2}b - \frac{3}{2}c + \frac{1}{2}d\right)x^3 + \left(a - \frac{5}{2}b + 2c - \frac{1}{2}d\right)x^2 + \left(-\frac{1}{2}a + \frac{1}{2}c\right)x + b \quad (6)$$



Так как бикубическая интерполяция представляет собой кубическую интерполяцию в двух измерениях, то можно, используя формулу кубической интерполяции (6), построить бикубическую интерполяционную формулу.

Представим, что известно 16 точек p_{ij} , с точкой начала координат в $(i-1, j-1)$, где i, j изменяются от 0 до 3. Тогда, можно интерполировать поверхность на участке $[0,0] \times [1,1]$, для этого интерполируются 4 колонки и потом интерполируются полученные результаты в горизонтальном направлении:

$$g(x, y) = f(x, f(y, p_{00}, p_{01}, p_{02}, p_{03}), f(y, p_{10}, p_{11}, p_{12}, p_{13}), f(y, p_{20}, p_{21}, p_{22}, p_{23}), f(y, p_{30}, p_{31}, p_{32}, p_{33})) \quad (7)$$

Перепишем формулу для бикубической интерполяции как многомерный полином:

$$g(x, y) = \sum_{i=0}^3 \sum_{j=0}^3 a_{ij} x^i y^j \quad (8)$$

С помощью этих значений для ij , вычисляются коэффициенты:

$$a_{00} = p_{11}$$

$$a_{01} = -\frac{1}{2}p_{10} + \frac{1}{2}p_{12}$$

$$a_{02} = p_{10} - \frac{5}{2}p_{11} + 2p_{12} - \frac{1}{2}p_{13}$$

$$a_{03} = -\frac{1}{2}p_{10} + \frac{3}{2}p_{11} - \frac{3}{2}p_{12} + \frac{1}{2}p_{13}$$

$$a_{10} = -\frac{1}{2}p_{01} + \frac{1}{2}p_{21}$$

$$a_{11} = \frac{1}{4}p_{00} - \frac{1}{4}p_{02} - \frac{1}{4}p_{20} + \frac{1}{4}p_{22}$$

$$a_{12} = -\frac{1}{2}p_{00} + \frac{5}{4}p_{01} - p_{02} + \frac{1}{4}p_{03} + \frac{1}{2}p_{20} - \frac{5}{4}p_{21} + p_{22} - \frac{1}{4}p_{23}$$

$$a_{13} = \frac{1}{4}p_{00} - \frac{3}{4}p_{01} + \frac{3}{4}p_{02} - \frac{1}{4}p_{03} - \frac{1}{4}p_{20} + \frac{3}{4}p_{21} - \frac{3}{4}p_{22} + \frac{1}{4}p_{23}$$

$$a_{20} = p_{01} - \frac{5}{2}p_{11} + 2p_{21} - \frac{1}{2}p_{31}$$

$$a_{21} = -\frac{1}{2}p_{00} + \frac{1}{2}p_{02} + \frac{5}{4}p_{10} - \frac{5}{4}p_{12} - p_{20} + p_{22} + \frac{1}{4}p_{30} - \frac{1}{4}p_{32}$$

$$a_{22} = p_{00} - \frac{5}{2}p_{01} + 2p_{02} - \frac{1}{2}p_{03} - \frac{5}{2}p_{10} + \frac{25}{4}p_{11} - 5p_{12} + \frac{5}{4}p_{13} +$$

$$+ 2p_{20} - 5p_{21} + 4p_{22} - p_{23} - \frac{1}{2}p_{30} + \frac{5}{4}p_{31} - p_{32} + \frac{1}{4}p_{33}$$

$$a_{23} = -\frac{1}{2}p_{00} + \frac{3}{2}p_{01} - \frac{3}{2}p_{02} + \frac{1}{2}p_{03} + \frac{5}{4}p_{10} - \frac{15}{4}p_{11} + \frac{15}{4}p_{12} -$$

$$-\frac{5}{4}p_{13} - p_{20} + 3p_{21} - 3p_{22} + p_{23} + \frac{1}{4}p_{30} - \frac{3}{4}p_{31} + \frac{3}{4}p_{32} - \frac{1}{4}p_{33}$$

$$a_{30} = -\frac{1}{2}p_{01} + \frac{3}{2}p_{11} - \frac{3}{2}p_{21} + \frac{1}{2}p_{31}$$

$$a_{31} = \frac{1}{4}p_{00} - \frac{1}{4}p_{02} - \frac{3}{4}p_{10} + \frac{3}{4}p_{12} + \frac{3}{4}p_{20} - \frac{3}{4}p_{22} - \frac{1}{4}p_{30} + \frac{1}{4}p_{32}$$

$$a_{32} = -\frac{1}{2}p_{00} + \frac{5}{4}p_{01} - p_{02} + \frac{1}{4}p_{03} + \frac{3}{2}p_{10} - \frac{15}{4}p_{11} + 3p_{12} - \frac{3}{4}p_{13} -$$

$$-\frac{3}{2}p_{20} + \frac{15}{4}p_{21} - 3p_{22} + \frac{3}{4}p_{23} + \frac{1}{2}p_{30} - \frac{5}{4}p_{31} + p_{32} - \frac{1}{4}p_{33}$$

$$a_{33} = \frac{1}{4}p_{00} - \frac{3}{4}p_{01} + \frac{3}{4}p_{02} - \frac{1}{4}p_{03} - \frac{3}{4}p_{10} + \frac{9}{4}p_{11} - \frac{9}{4}p_{12} +$$

$$+\frac{3}{4}p_{13} + \frac{3}{4}p_{20} - \frac{9}{4}p_{21} + \frac{9}{4}p_{22} - \frac{3}{4}p_{23} - \frac{1}{4}p_{30} + \frac{3}{4}p_{31} - \frac{3}{4}p_{32} + \frac{1}{4}p_{33}$$



Другим методом увеличения разрешающей способности изображения является метод бикубического сплайна.

Сплайн – функция, которая вместе с несколькими производными непрерывна на всем заданном отрезке $[a, b]$, а на каждом частичном отрезке $[x_i, x_{i+1}]$ в отдельности является некоторым алгебраическим многочленом.

Степенью сплайна называется максимальная по всем частичным отрезкам степень многочленов, а дефектом сплайна – разность между степенью сплайна и порядком наивысшей непрерывной на $[a, b]$ производной.

На практике наиболее часто используются кубические сплайны $S_3(x)$ – сплайны третьей степени с непрерывной, по крайней мере, первой производной. При этом величина $m_i = S'_3(x_i)$, называется наклоном сплайна в точке (узле) x_i .

Разобьем отрезок $[a, b]$ на N равных отрезков $[x_i, x_{i+1}]$, где $x_i = a + ih$, $i = 0, 1, \dots, N-1$, $x_N = b$, $h = (b-a)/N$.

Если в узлах x_i, x_{i+1} заданы значения f_i, f_{i+1} , которые принимает кубический сплайн, то на частичном отрезке $[x_i, x_{i+1}]$ он принимает вид:

$$S_3(x) = \frac{(x_{i+1}-x)^2(2(x-x_i)+h)}{h^3} f_i + \frac{(x-x_i)^2(2(x_{i+1}-x)+h)}{h^3} f_{i+1} + \frac{(x_{i+1}-x)(x-x_i)}{h^2} m_i + \frac{(x-x_i)^2(x-x_{i+1})}{h^2} m_{i+1} \quad (9)$$

В самом деле, это легко проверить, рассчитав $S_3(x)$ и $S'_3(x)$ в точках x_i, x_{i+1} .

Можно доказать, что если многочлен третьей степени, принимает в точках x_i, x_{i+1} значения f_i, f_{i+1} и имеет в этих точках производные, соответственно, m_i, m_{i+1} , то он совпадает с многочленом (9).

Таким образом, для того, чтобы задать кубический сплайн на отрезке, необходимо задать значения $f_i, m_i, i = 0, 1, \dots, N$ в $N+1$ в узле x_i .

Кубический сплайн, принимающий в узлах те же значения f_i , что и некоторая функция, называется интерполяционным и служит для аппроксимации функции f на отрезке $[a, b]$ вместе с несколькими производными.

Существует несколько способов задания наклонов интерполяционного кубического сплайна.

Способ 1 (упрощенный):

Положим:

$$m_i = \frac{f_{i+1} - f_{i-1}}{2h}, \quad i = 1, 2, \dots, N-1, \quad (10)$$

$$m_0 = \frac{4f_1 - f_2 - 3f_0}{2h}, \quad m_N = \frac{3f_N - f_{N-2} - 3f_{N-1}}{2h}$$

Данные формулы являются формулами численного дифференцирования второго порядка точности относительно шага $h = (b-a)/N$.

Способ 2:

Если у нас имеются значения f'_i производной f_i в узлах x_i , то полагаем $m_i = f'_i, i = 0, 1, \dots, N$.

Способ 3 (глобальный):

Пусть $S''_3(x_i+0)$ – значение $S''_3(x)$ в узле x_i справа, его мы найдем из выражения (9), а $S''_3(x_i-0)$ – значение $S''_3(x)$ в узле x_i слева – оно находится из со-



ответствующего выражения $S_3(x)$ на частичном отрезке $[x_i, x_{i+1}]$, которое получается из (9) заменой i на $i-1$.

Тогда получим:

$$S''_3(x_i+0) = \frac{-4m_i}{h} - \frac{2m_{i+1}}{h} + 6 \frac{f_{i+1} - f_i}{h^2};$$

$$S''_3(x_i-0) = \frac{2m_{i-1}}{h} + \frac{4m_i}{h} - 6 \frac{f_i - f_{i-1}}{h^2} \quad (11)$$

Потребуем непрерывность $S''_3(x)$ в узлах:

$$S''_3(x_i-0) = S''_3(x_i+0), \quad i = 1, 2, \dots, N-1.$$

Тогда получим систему линейных алгебраических уравнений относительно наклонов:

$$m_{i-1} + 4m_i + m_{i+1} = \frac{3(f_{i+1} - f_{i-1})}{h}, \quad i = 1, 2, \dots, N-1. \quad (12)$$

Так как система содержит $N+1$ неизвестных, необходимо задать два дополнительных условия, называемые краевыми.

Приведем три варианта задания краевых условий:

$$1) \text{ в случае, когда известны } m_N = f'_N \text{ задаем: } m_0 = f'_0, m_N = f'_N \quad (13)$$

2) производные f'_0, f'_N аппроксимируем формулами численного дифференцирования третьего порядка точности:

$$m_0 = \frac{1}{6h} (-11f_0 + 18f_1 - 9f_2 + 2f_3) \quad (14)$$

$$m_N = \frac{1}{6h} (11f_N - 18f_{N-1} + 9f_{N-2} - 2f_{N-3})$$

3) Иногда бывают известны значения f''' на концах отрезка $[a, b]$, т.е. величины $f'''_0 = f'''(a), f'''_N = f'''(b)$. Тогда требования

$S''_3(a) = f'''_0, S''_3(b) = f'''_N$, приводят к краевым условиям:

$$m_0 = \frac{-m_1}{2} + \frac{3f_1 - f_0}{2h} - \frac{h}{4} f'''_0 \quad (15)$$

$$m_N = \frac{-m_{N-1}}{2} + \frac{3f_N - f_{N-1}}{2h} - \frac{h}{4} f'''_N$$

Условия (13)-(15) можно комбинировать, т.е. выбирать их независимо в левом и правом узлах.

Система (12) при всех рассмотренных краевых условиях имеет единственное решение, которое можно найти с помощью методов прогонки и итераций.

Таким образом, решая систему (12) при выбранных краевых условиях, находим наклоны $m_i, i = 0, 1, \dots, N$, во всех узлах. Затем по формуле (9) задаем сплайн на каждом частичном отрезке $[x_i, x_{i+1}], i = 0, 1, \dots, N-1$. Построенный данным глобальным способом сплайн $S_3(x)$ имеет дефект не больше единицы, так как этот сплайн обладает на отрезке $[a, b]$ непрерывной второй производной $S''_3(x)$.

Метод кубических сплайнов выдает более четкое изображение по сравнению с бикубической интерполяцией, поскольку он способен воспроизвести четкую и гладкую границу без ступенек.

Вышеизложенные и другие методы применяются во многих программах редактирования изображений, драйверах принтеров и встроенной интерполяции камер. В рамках статьи были рассмотрены несколько решений по увеличе-



нию разрешения способности изображений, таких как Enlarge, Smilla Enlarger, PhotoZoom Pro.

В таблице 1 приведена сравнительная характеристика данных программ.

Таблица 1

Программа	Вид коммерции	Метод обработки	Графический интерфейс
Enlarge	бесплатная	Бикубическая интерполяция с применением фильтрации	Отсутствует
PhotoZoom Pro	платная	Различные вариации метода бикубического сплайна с применением фильтрации	Ориентирован на продвинутого пользователя
Smilla Enlarger	бесплатная	Бикубическая интерполяция с применением фильтрации	Простой и удобный

Однозначно выделить лучшее среди представленных коммерческих решений довольно сложно, поскольку многое зависит от типа исходных изображений, однако, в целом более качественную обработку обеспечивает PhotoZoom Pro по сравнению с другими решениями представленными в таблице 1. Среди бесплатных же программ по качеству получаемых изображений заметно выделяется Enlarge, однако, по удобству применения Enlarge проигрывает утилите SmillaEnlarge.

Литература

1. Интерполяция цифрового изображения: Учебник цифровой фотографии. - 2015. (<http://www.cambridgeincolour.com/ru/tutorials/image-interpolation.htm>)
2. Гонсалес, Р. Обработка цифровых изображений [Текст]: монография / Р. Гонсалес, Р. Вудс. – М: Техносфера. – 2005. – 114 с.
3. Keys, R. Cubic convolution interpolation for digital image processing.
4. Acoustics, Speech and Signal Processing [Текст]: монография / R. Keys, - IEEE Transactions on, 29(6): 1981. – 1153-1160 с.
5. Боровиков, В.П. STATISTICA. Искусство анализа данных на компьютере: для профессионалов (2-е издание) [Текст]: монография / В.П. Боровиков. – СПб: Питер. – 2003. – 688 с.
6. Волков, Е.А. Численные методы [Текст]: монография / Е.А. Волков. – М: «Наука». – 1987.