УДК 004.627

А.Н. Пчелинцев, М.М. Дружинин

РЕШЕНИЕ ОБРАТНОЙ ЗАДАЧИ ПРИ МОДЕЛИРОВАНИИ СЛУЧАЙНЫХ МАРКОВСКИХ ПРОЦЕССОВ ПРИМЕНИТЕЛЬНО К СЖАТИЮ ИЗОБРАЖЕНИЙ*

В течение последних 10 лет в рамках компьютерной графики бурно развивается совершенно новая область – алгоритмы архивации изображений. Появление этой области обусловлено тем, что изображения – своеобразный тип данных. Мы можем легко заметить, что изображение, в отличие, например, от текста, обладает избыточностью.

При использовании некоторой системы цветопредставления каждый пиксел представляет собой запись (структуру), полями которой являются компоненты цвета. Самой распространенной является система RGB, в

* Работа выполняется под руководством проф. В.Ф. Першина и доцента С.А. Васильева.

которой цвет представлен значениями красной, зеленой и синей компонент. Эта модель также может рассматриваться как совокупность трех цветовых плоскостей, каждая из которых хранит информацию о своей компоненте цвета. Заметим, что RGB-модель используется в данной работе.

Возьмем одну из цветовых плоскостей, например, красную (R). Обозначим через S^* вектор-строку, который формируется из данных этой цветовой плоскости по следующему алгоритму: цветовая плоскость принимается как вектор-строка, составленная из строк этой плоскости, каждая из которых представлена набором целых чисел (байт). Эта строка разбивается на совокупность блоков длины m.

Рассмотрим j-й блок. Каждый элемент этого блока ставится в соответствующую позицию вектора S^* (причем, целочисленные данные преобразуются в формат с плавающей запятой).

Далее при моделировании будем использовать аппарат случайных марковских процессов. Заметим, что вектором $S^{(n)}$ можно представить состояние некой системы в конечный момент времени n. Состояние системы в произвольный момент времени k может быть определено из формулы:

$$S^{(k)} = S^{(k-1)} P, \tag{1}$$

где P – квадратная матрица порядка m переходных вероятностей, на которую наложено условие, что сумма элементов в каждой строке равна единице, $k = 1, ..., n$.

Можно подобрать такой вид матрицы P и такое количество шагов n, что из начального состояния $S^{(0)}$ по формуле (1) мы получим состояние $S^{(n)}$, близкое к S^*.

Вектор начального состояния $S^{(0)}$ будем искать в следующем виде: первый элемент данного вектора ненулевой, остальные равны нулю. Здесь появляется избыточность, на которой и основано сжатие. Так как процесс марковский, то сумма элементов вектора состояния постоянна во времени. Из этого следует, что первый элемент вектора $S^{(0)}$ должен быть равен сумме элементов вектора S^*, приближение $S^{(n)}$ которого мы ищем.

Рассматриваемый метод является сжатием с потерями. Но если добиться наибольшего отклонения значений вектора $S^{(n)}$ от S^* метода, меньше половины значащего разряда, то мы получим сжатие без потерь.

При численном моделировании было установлено, что наилучшее приближение достигается, если матрица P является треугольной (верхний треугольник) и ее элементы рассчитываются по алгоритму: сначала каждому элементу p_{ij} из верхнего треугольника присваивается значение p exp($-xxpp$), где $x = 2(j - i)/m$, а p задает вид матрицы. После чего каждый элемент строки делится на сумму элементов строки, в которой он содержится. Деление произведено для выполнения условия Маркова: сумма элементов каждой строки матрицы P равна единице.

Перейдем к постановке задачи оптимизации. Критерием (минимизируемой функцией) здесь является точность приближения, которая может быть выражена как норма разности векторов $S^{(n)}$ и S^*. Варьируемые параметры: p и n. В ходе численных экспериментов были установлены границы изменения параметров: для n – [1; $2m$], для p – [–2; 2]. Параметр p методом золотого сечения при фиксированном n, который ищется методом сканирования.

Хотелось бы заметить, что вид матрицы P существенно определяет вид конечного состояния $S^{(n)}$. Поэтому разработка вида данной матрицы может дать более точные результаты.

Программная реализация была выполнена на языке С++. Ее результатами являются следующие изображения, показанные на рис. 1 – 3.

На рис. 1 представлено исходное изображение. На рис. 2 изображение – сжатое со степенью сжатия 40 %, а на рис. 3 – со степенью 25 %.

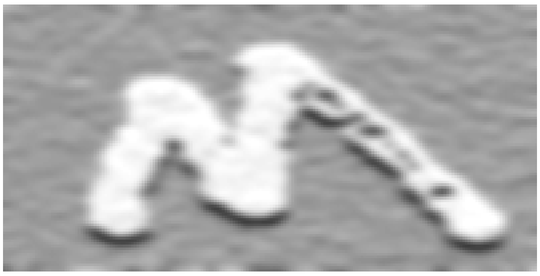

Рис. 1

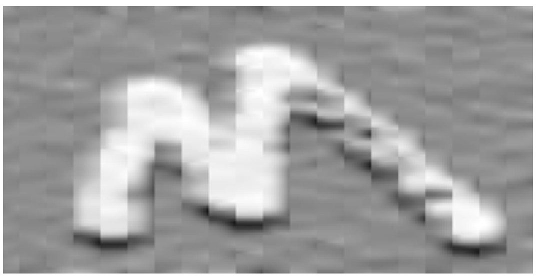

Рис. 2

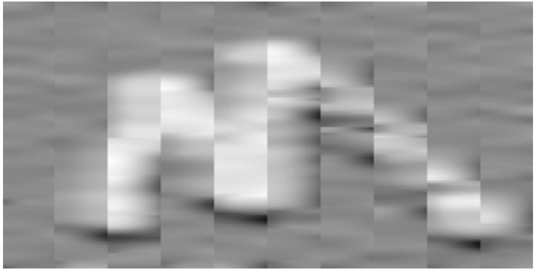

Рис. 3

Так как алгоритм оперирует данными в формате с плавающей запятой, то его можно применить для сжатия других потоков данных.

Список литератруы

1 Один из вариантов решения обратной задачи при моделировании случайных марковских процессов / А.Н. Пчелинцев, С.В. Першина, В.Г. Однолько, В.Ф. Першин // Математические методы в технике и технологиях – ММТТ-18 : сб. тр. XVIII Междунар. науч. конф. В 10 т. Т. 8. Секции 10, 12. Казань : Изд-во Казанского гос. технол. ун-та, 2005. С. 178 – 179.

2 http://docs.luksian.com/programming/theory/imgcomp.

Кафедра «Прикладная механика и сопротивление материалов»