



Рис. 2. Спектральные кривые для массива двух резонаторов с четырьмя отверстиями (расстояние между резонаторами – 350нм)

### Литература

1. Казанский Н.Л., Серафимович П.Г. Coupled-resonator optical waveguides for temporal integration of optical signals [Электронный ресурс] // Optics Express. – 2014. Режим доступа: <http://www.opticsinfobase.org/oe/fulltext.cfm?uri=oe-22-11-4004&id=286668>
2. Photonic Crystals: Molding the Flow of Light. Second Edition/ John D. Joannopoulos, Steven G. Johnson, Joshua N. Winn, Robert D. Meade. – Princeton University Press, 2008. – 305 p.
3. Band diagram, resonant modes, and transmission in a holey waveguide [Электронный ресурс] // Meep Tutorial. – 2010. Режим доступа: [http://ab-initio.mit.edu/wiki/index.php/Meep\\_Tutorial/Band\\_diagram,\\_resonant\\_modes,\\_and\\_transmission\\_in\\_a\\_holey\\_waveguide](http://ab-initio.mit.edu/wiki/index.php/Meep_Tutorial/Band_diagram,_resonant_modes,_and_transmission_in_a_holey_waveguide)

А.А. Смирнов

## КЛАССИФИКАЦИЯ ТЕКСТУРНЫХ ИЗОБРАЖЕНИЙ С ИСПОЛЬЗОВАНИЕМ НЕЙРОННЫХ СЕТЕЙ

(Самарский государственный аэрокосмический университет имени академика С.П. Королева (национальный исследовательский университет))

Задача классификации изображения заключается в отнесении этого изображения к одному из нескольких классов. Необходимо заметить, что в этой задаче множество классов, к которым может быть отнесено изображение, заранее



известно. Примером таких задач могут быть, например, медицинские задачи, в которых необходимо определить исход заболевания.

Нейронные сети являются универсальным средством аппроксимации функций, что позволяет использовать их в задаче классификации. Нейронные сети можно рассматривать как современные вычислительные системы, которые преобразуют информацию по образу процессов, происходящих в мозгу человека. Обработываемая информация имеет численный характер, что позволяет использовать нейронную сеть в качестве модели объекта с совершенно неизвестными характеристиками.

Для построения классификатора необходимо определить, какие параметры влияют на принятие решения о том, к какому классу принадлежит образец. При этом могут возникнуть две проблемы. Во-первых, если количество параметров мало, то может возникнуть ситуация, при которой один и тот же набор исходных данных соответствует примерам, находящимся в разных классах. Тогда невозможно обучить нейронную сеть, и система не будет корректно работать (невозможно найти минимум, который соответствует такому набору исходных данных). Во-вторых, исходные данные обязательно должны быть непротиворечивы. Для решения этой проблемы необходимо увеличить размерность пространства признаков (количество компонент входного вектора, соответствующего образцу). Но при увеличении размерности пространства признаков может возникнуть ситуация, когда число примеров может стать недостаточным для обучения сети, и она вместо обобщения просто запомнит примеры из обучающей выборки и не сможет корректно функционировать. Поэтому при определении признаков необходимо найти компромисс с их количеством.

### **1. Вычисление текстурных признаков на основе матрицы вхождений**

Для описания изображений с использованием текстурных признаков был реализован метод, основанный на матрицах вероятностного распределения яркости на изображении. Такой подход дает большой набор признаков, и можно предполагать, что, по крайней мере, один из них отражает текстурные изменения между классами. Элементами матрицы являются вероятности второго порядка, которые учитывают как уровни яркости, так и относительное расположение пикселей на изображении.

Матрица вхождений – это двумерный массив  $C$ , в котором индексы строк и столбцов образуют множество  $V$  допустимых на изображении значений пикселей. Например, для полутоновых изображений  $V$  может быть множеством допустимых значений интенсивности (яркости). Значение  $C(i, j)$  указывает, сколько раз значение  $i$  встречалось на изображении в некотором заданном пространственном отношении со значением  $j$ . Например, в качестве пространственного отношения может быть выбрано «значение  $i$  является правым соседом для значения  $j$ ».

Обозначим через  $d$  вектор перемещения  $(dr, dc)$ , где  $dr$  соответствует перемещению в направлении строк (вниз по изображению), а  $dc$  перемещение в



направлении столбцов (вправо по изображению). Полутоновая матрица вхождений  $C_d$  для изображения  $I$  определяется выражением:

$$C_d[i, j] = |\{[r, c] | I[r, c] = i \wedge I[r + dr, c + dc] = j\}|$$

Одна из разновидностей стандартной полутоновой матрицы вхождений называется нормированной полутоновой матрицей вхождений  $N_d$  и определяется выражением:

$$N_d[i, j] = \frac{C_d[i, j]}{\sum_i \sum_j C_d[i, j]}$$

Вычисленные, согласно этому выражению значения вхождений принадлежат диапазону  $[0, 1]$ . Их можно рассматривать в качестве значений вероятности в матрице большего размера.

Для того, чтобы устранить зависимость элементов матрицы от поворота изображения для вычисления признаков использовалась матрица  $C_d[i, j]$ , которая вычислялась как среднее четырех матриц, вычисленных при каждом из четырех углов поворота ( $0^\circ$ ,  $45^\circ$ ,  $90^\circ$  и  $180^\circ$ ):

$$C_d = (C_{d,0} + C_{d,d} + C_{0,d} + C_{0,-d}) / 4$$

Таким образом, матрица позволяет оценить преимущественное расположение отсчетов на изображении и их коррелированность.

Матрицы вхождений представляют свойства текстуры, но они не удобны для непосредственного применения при анализе изображений. Вместо этого матрицы вхождений используются для вычисления числовых характерных признаков, которые могут служить более компактным представлением текстуры. На основе нормированной матрицы вхождений можно вычислить следующие характерные признаки:

1. Суммарное среднее:  $F_1 = \sum_i m_i p_i$
2. Инерция:  $F_2 = \sum_i \sum_j (i - j)^2 N_d(i, j)$
3. Второй угловой момент  $F_3 = \sum_i \sum_j N_d^2(i, j)$
4. Суммарная корреляция  $F_4 = \sum_i \sum_j m_i m_j$
5. Энтропия:  $F_5 = -\sum_i \sum_j \ln(N_d(i, j)) N_d(i, j)$
6. Корреляция:  $F_6 = \sum_i \sum_j (i - M_x)(j - M_x) N_d(i, j)$
7. Затенение:  $F_7 = \sum_i \sum_j (i + j - 2M_x)^3 N_d(i, j)$
8. Контраст:  $F_8 = \sum_i \sum_j |i - j| N_d(i, j)$
9. Суммарная энтропия:  $F_9 = \sum_i \ln(p_i) p_i$

где  $m_i = \sum_j j N_d(i, j)$ ,  $p_i = \sum_j N_d(i, j)$ ,  $M_x = \sum_i i p_i$ .



## 2. Обучение нейронной сети

Для обучения многослойного персептрона используется алгоритм обратного распространения ошибки. Этот алгоритм основывается на коррекции ошибки. Обучение методом обратного распространения ошибки предполагает два прохода по всем слоям сети: прямого и обратного. При прямом проходе входной вектор подается на входной слой нейронной сети, после чего распространяется по сети от слоя к слою. В результате генерируется набор выходных сигналов, который и является фактической реакцией сети на данный входной образ. Во время прямого прохода все синаптические веса сети фиксированы. Во время обратного прохода все синаптические веса настраиваются в соответствии с правилом коррекции ошибок, а именно: фактический выход сети вычитается из желаемого, в результате чего формируется сигнал ошибки. Этот сигнал впоследствии распространяется по сети в направлении, обратном направлению синаптических связей. Отсюда и название – алгоритм обратного распространения ошибки. Синаптические веса настраиваются с целью максимального приближения выходного сигнала сети к желаемому.

Алгоритм обратного распространения ошибки следующий:

1. Инициализировать синаптические веса маленькими случайными значениями.
2. Выбрать очередную обучающую пару из обучающего множества; подать входной вектор на вход сети.
3. Вычислить выход сети.
4. Вычислить разность между выходом сети и требуемым выходом (целевым вектором обучающей пары).
5. Подкорректировать веса сети для минимизации ошибки.
6. Повторять шаги с 2 по 5 для каждого вектора обучающего множества до тех пор, пока ошибка на всем множестве не достигнет приемлемого уровня.

Целью обучения сети алгоритмом обратного распространения ошибки является такая подстройка ее весов, чтобы приложение некоторого множества входов приводило к требуемому множеству выходов. Для краткости эти множества входов и выходов будут называться векторами. При обучении предполагается, что для каждого входного вектора существует парный ему целевой вектор, задающий требуемый выход. Вместе они называются обучающей парой. Сеть обучается на многих парах.

## 3. Результаты классификации текстур

Был исследован набор из 49 изображений размерами 512 на 512 пикселей. В этом наборе было 7 представителей от каждого класса. Таким образом, было доступно 7 классов изображений по 7 изображений на каждый класс.

В ходе эксперимента выбиралось одно изображение для тестовой выборки и 48 для обучающей. Эта процедура повторялась 49 раз для каждой нейронной сети. После каждого формирования проводилась классификация, и фиксировалось число верно проклассифицированных изображений из каждого класса. Так же фиксировалось количество неверно проклассифицированных изображений, которые либо не попали в свой класс, либо попали в чужой.



В таблице 1 представлена матрица ошибок классификации текстурных изображений с использованием нейронной сети. Она показывает вероятность попадания изображения из класса  $i$  в класс  $j$ . На главной диагонали этой матрицы располагаются вероятности правильной классификации.

Таблица 1 – Матрица ошибок классификации текстурных изображений

	bark	brick	bubbles	grass	leather	pigskin	rafia
bark	1	0	0	0	0	0	0
brick	0	1	0	0	0	0	0
bubbles	0	0	0.5714	0	0.1429	0	0
grass	0.2857	0	0	1	0	0	0
leather	0	0	0.2857	0	0.8571	0	0
pigskin	0	0	0.1429	0	0	1	0.1429
rafia	0	0	0	0	0	0	1

### Литература

1. Naralick, R (1979). Statistical and structural approaches to texture. Proceedings of the IEEE 67, 786–804.
2. Шапиро Л., Стокман Дж. Компьютерное зрение / Пер. с англ. – М.: БИНОМ. Лаборатория знаний, 2006. – 752 с.
3. Хайкин С. Нейронные сети: полный курс, 2-е изд., испр. / Пер. с англ. – М.: ООО «И.Д. Вильямс», 2006. – 1104 с.

Я.В. Соловьева, О.В. Борисова

### АВТОМАТИЗИРОВАННАЯ СИСТЕМА УПРАВЛЕНИЯ МНОГОПОЛЬЗОВАТЕЛЬСКИМИ РЕЛЯЦИОННЫМИ БАЗАМИ ДАННЫХ

(Самарский государственный аэрокосмический университет имени академика С.П. Королева (национальный исследовательский университет))

С ростом популярности систем управления базами данных (СУБД) в 70-80-х годах появилось множество различных моделей данных. Первые СУБД использовали иерархическую модель данных. Для чтения данных из иерархической базы данных требовалось перемещаться по записям, за один раз переходя на одну запись вверх, вниз или в сторону. Если структура данных оказывалась сложнее, чем обычная иерархия, простота структуры иерархической базы данных становилась её недостатком. В связи с этим была разработана новая сетевая модель данных. Разница между иерархической моделью данных и сетевой состоит в том, что в иерархических структурах запись-потомок должна иметь в точности одного предка, а в сетевой структуре данных у потомка может иметься любое число предков. Сетевая БД состоит из набора экземпляров определенного типа записи и набора экземпляров определенного типа связей между