

Лабораторная работа №1

Тема: Сжатие графической информации линейной рециркуляционной сетью.

Цель: Ознакомиться, проанализировать и получить навыки реализации модели линейной рециркуляционной сети для задачи сжатия графической информации.

Для лабораторной работы в качестве данных исходного растрового изображения можно использовать файлы следующих форматов:

- BMP;
- TIFF;
- PNG;
- JPG.

Можно использовать как цветное, так и чёрно-белое изображение, где значения интенсивности цвет пиксела задаются $\mathbf{R}, \mathbf{G}, \mathbf{B}$ на множестве шкал \mathbf{S} (например $\mathbf{S} = \{\mathbf{R}, \mathbf{G}, \mathbf{B}\}$). Исходное изображение $\mathbf{h} \times \mathbf{w}$ (рекомендуемый размер – не менее 256×256 (или картинка с таким же числом пикселей)) разбивается на \mathbf{L} прямоугольников размерностью $\mathbf{n} \times \mathbf{m}$ ($\mathbf{n} * \mathbf{m} = \mathbf{N}/|\mathbf{S}|$) ($\mathbf{n} \leq \mathbf{h}$, $\mathbf{m} \leq \mathbf{w}$), где значения \mathbf{n}, \mathbf{m} может указать пользователь. Все прямоугольники должны полностью покрывать исходное изображение и не выходить за его границы, может допускаться перекрытие прямоугольников, которое может задать пользователь.

Пусть \mathbf{C}_i обозначает одно из значений $\mathbf{R}, \mathbf{G}, \mathbf{B}$ ($i \in \mathbf{S}$) и принимает значения в диапазоне $0 \leq \mathbf{C}_i \leq 255 = \mathbf{C}_{\max}$. Для того, чтобы использовать значения цвета пиксела исходного изображения для дальнейшей обработки, следует выполнить преобразование:

$$\mathbf{c}_i(jk) = (2 * \mathbf{C}_i(jk) / \mathbf{C}_{\max}) - 1, \quad (1)$$

где $\mathbf{c}_i(jk)$ – это значение цвета пиксела для дальнейших преобразований (jk – координаты пикселя).

Так преобразуются все значения пикселей для всех прямоугольников. Значения цвета пикселей из прямоугольника \mathbf{q} собираются в свой (эталонный) вектор $\mathbf{X}_0(\mathbf{q})$, компоненты которого вычисляются в соответствии с выражением $\mathbf{X}_0(\mathbf{q})_{i+|\mathbf{S}|*(j+k*\mathbf{n})} = \mathbf{c}(\mathbf{q})_i(jk)$, где $j \leq \mathbf{n}$, $k \leq \mathbf{m}$. Т. е. число элементов $\mathbf{X}_0(\mathbf{q})$ будет равно \mathbf{N} , где $\mathbf{X}_0(\mathbf{q})$ – это вектор размера $1 \times \mathbf{N}$, а его значения после преобразований уже будут $|\mathbf{X}_0(\mathbf{q})_k| \leq 1$.

Пример 1:

$$\begin{matrix} & 0.1 & 0.5 & 0.2 \\ \mathbf{c}(\mathbf{q}) = & 0.3 & -0.2 & 0.6, \\ & 0.7 & -0.7 & 0.9 \end{matrix}$$

тогда $\mathbf{X}_0(\mathbf{q}) = [0.1, 0.5, 0.2, 0.3, -0.2, 0.6, 0.7, -0.7, 0.9]$.

Далее следует использовать матрицу весов \mathbf{W} для первого слоя сети размерностью $\mathbf{N} \times \mathbf{p}$, где \mathbf{p} – это заданное пользователем число нейронов второго слоя (как правило $\mathbf{p} \leq 2 \times \mathbf{N}$). Значения весов должны быть $|\mathbf{W}_{ij}| \leq 1$. Матрица весов на втором слое $\mathbf{W}' = [\mathbf{W}]^T$.

Производится обучение последовательно на выборке из \mathbf{L} эталонных образов длиной \mathbf{N} . Принимается $\mathbf{X}(\mathbf{i}) = \mathbf{X}_0(\mathbf{i})$. Следует произвести следующие преобразования:

$$\mathbf{Y}(\mathbf{i}) = \mathbf{X}(\mathbf{i}) * \mathbf{W} \quad (2)$$

$$\mathbf{X}'(\mathbf{i}) = \mathbf{Y}(\mathbf{i}) * \mathbf{W}' \quad (3)$$

$$\Delta \mathbf{X}(\mathbf{i}) = \mathbf{X}'(\mathbf{i}) - \mathbf{X}(\mathbf{i}) \quad (4)$$

Для корректировки весов на втором слое следует вычислить значения согласно формуле:

$$\mathbf{W}'(\mathbf{t} + 1) = \mathbf{W}'(\mathbf{t}) - \alpha' * [\mathbf{Y}(\mathbf{i})]^T * \Delta \mathbf{X}(\mathbf{i}), \quad (5)$$

где $0 < \alpha' \leq 0.01$ – коэффициент обучения, указываемый пользователем либо вычисляемый согласно выражению для расчета адаптивного шага:

$$\alpha' = 1/(\mathbf{Y}(\mathbf{i}) * \mathbf{Y}(\mathbf{i})^T),$$

Для обучения нейронов первого слоя следует вычислить:

$$\mathbf{W}(\mathbf{t} + 1) = \mathbf{W}(\mathbf{t}) - \alpha * [\mathbf{X}(\mathbf{i})]^T * \Delta \mathbf{X}(\mathbf{i}) * [\mathbf{W}'(\mathbf{t})]^T, \quad (6)$$

где $0 < \alpha \leq 0.01$ – коэффициент обучения, указываемый пользователем ($\alpha = \alpha'$) либо вычисляемый согласно выражению для расчета адаптивного шага:

$$\alpha = 1/(\mathbf{X}(\mathbf{i}) * \mathbf{X}(\mathbf{i})^T),$$

Нормализация может быть осуществлена согласно выражениям:

$$\mathbf{W}_{ij} = \mathbf{W}_{ij} / \|\mathbf{W}\|^T j|,$$

$$\mathbf{W}'_{ji} = \mathbf{W}'_{ji} / \|\mathbf{W}'\|^T i|.$$

Максимальную допустимую ошибку \mathbf{e} задаёт пользователь, где $0 < \mathbf{e} \leq 0.1 * \mathbf{p}$ ($\alpha \leq \mathbf{e}$). Суммарная среднеквадратическая ошибка для прямоугольника \mathbf{q} в процессе обучения высчитывается по формуле:

$$\mathbf{E}(\mathbf{q}) = \sum \Delta \mathbf{X}(\mathbf{q})_i * \Delta \mathbf{X}(\mathbf{q})_i, \text{ где } 1 \leq i \leq \mathbf{N} \quad (7)$$

Суммарная ошибка для обучающей выборки равна:

$$\mathbf{E} = \sum \mathbf{E}(\mathbf{q}), \text{ где } 1 \leq \mathbf{q} \leq \mathbf{L} \quad (8)$$

$\mathbf{E}(\mathbf{i} + 1)$ не вычисляется до тех пор, пока не произойдёт вычисление $\mathbf{E}(\mathbf{i})$. Т.е. вычисление по формулам (5) – (7) происходит \mathbf{L} раз.

Если $\mathbf{E} > \mathbf{e}$, тогда повторять действия, описанные в формулах (2) – (8), до тех пор, пока $\mathbf{E} \leq \mathbf{e}$.

Далее следует производить вычисления (сжатие и восстановление) для каждого эталонного прямоугольника по формулам (2) и (3), где $\mathbf{X}(\mathbf{i}) = \mathbf{X}_0(\mathbf{i})$, а

W и W' – веса, полученные в результате обучения. После восстанавливаются значения пикселей по формуле обратной формуле (1):

$$u_k = C_{\max} * (X'(i)_k + 1) / 2, \quad (9)$$

где u_k – значение пикселя для восстановленного изображения.

После восстановления пикселей собирается изображение из L элементов. Для этого производятся преобразования обратные тем, которые приведены в Примере 1. Полученное изображение выводится на экран. Так же требуется вывести результаты следующих вычислений:

- достигнутая ошибка при обучении;
- матрица весов на первом слое (для отладки);
- матрица весов на втором слое (для отладки);
- количество пройденных обучаемых шагов.

Коэффициент сжатия Z , при условии представления с одинаковой точностью всех числовых параметров:

$$Z = (N * L) / ((N + L) * p + 2), \quad (10)$$

В отчёте привести графики и таблицы следующих зависимостей:

- числа итераций обучения от коэффициента сжатия Z (для фиксированного изображения и параметров);
- числа итераций обучения для разных изображений (для фиксированных параметров и Z);
- числа итераций от e (остальные параметры фиксированы);
- числа итераций от α (остальные параметры фиксированы).

Задания:

1. Реализовать модель линейной рециркуляционной сети.
2. Реализовать модель линейной рециркуляционной сети с адаптивным шагом обучения.
3. Реализовать модель линейной рециркуляционной сети с нормированными весами.
4. Реализовать модель линейной рециркуляционной сети с адаптивным шагом обучения с нормированными весами.
5. Реализовать модель линейной рециркуляционной сети.
6. Реализовать модель линейной рециркуляционной сети с адаптивным шагом обучения.
7. Реализовать модель линейной рециркуляционной сети с нормированными весами.
8. Реализовать модель линейной рециркуляционной сети с адаптивным шагом обучения с нормированными весами.
9. Реализовать модель линейной рециркуляционной сети.
10. Реализовать модель линейной рециркуляционной сети с адаптивным шагом обучения.
11. Реализовать модель линейной рециркуляционной сети с нормированными весами.
12. Реализовать модель линейной рециркуляционной сети с адаптивным

- шагом обучения с нормированными весами.
13. Реализовать модель линейной рециркуляционной сети.
 14. Реализовать модель линейной рециркуляционной сети с адаптивным шагом обучения.
 15. Реализовать модель линейной рециркуляционной сети с нормированными весами.
 16. Реализовать модель линейной рециркуляционной сети с адаптивным шагом обучения с нормированными весами.

Литература:

1. PCA.rtf.
2. Головки В.А. Нейроинтеллект: теория и применение. Книга 1. Организация и обучение нейронных сетей с прямыми и обратными связями. Брест, - БПИ; 1999.
3. Головки В.А. Нейроинтеллект: теория и применение. Книга 2. Самоорганизация, отказоустойчивость и применение нейронных сетей. Брест, - БПИ; 1999.
4. Осовский С. Нейронные сети для обработки информации. М., - "Финансы и статистика"; 2002.