Grundlagen der digitalen Bildverarbeitung, Übung 3

Sven Köppel, 13.05.2011

Nr. 1 DFT-Identität

Definition der DFT aus der Vorlesung: $S(k)=\sum_{(n=0)}^{(N-1)} s(n) e^{-j\frac{2\pi}{N}kn}$, $k\in\{0,\dots,N-1\}, s(k)\in\mathbb{R}$

Zu zeigen: $S(l)=S(N-l)^*$

Dazu berechne

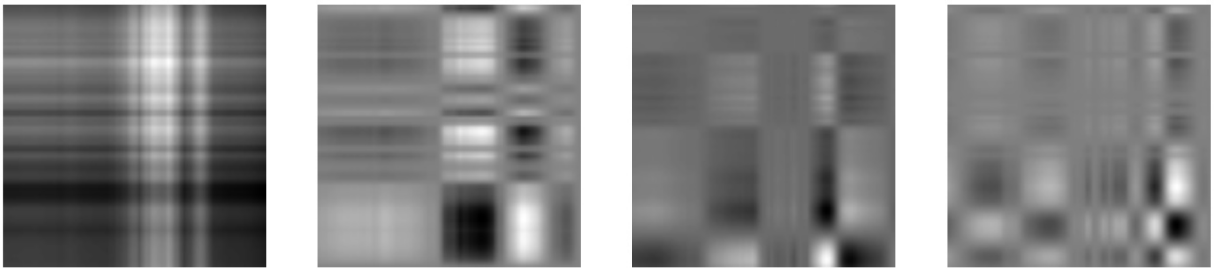
$$S(N-1)^*=\left(\sum_{(n=0)}^{(N-1)} s(n) e^{-j\frac{2\pi}{N}n(N-l)}\right)^*=\left(\sum_{(n=0)}^{(N-1)} s(n)\underbrace{e^{-j2\pi n}}_{=1} e^{+j\frac{2\pi}{N}nl}\right)^*=\sum_{(n=0)}^{(N-1)}\underbrace{s(n)^*}_{=s(n)} e^{-j\frac{2\pi}{N}nl}$$

$$=\sum_{(n=0)}^{(N-1)} s(n) e^{-j\frac{2\pi}{N}nl}=S(l)$$

qed.

Nr. 2 Entwicklung nach vollständigem Orthonormalsystem

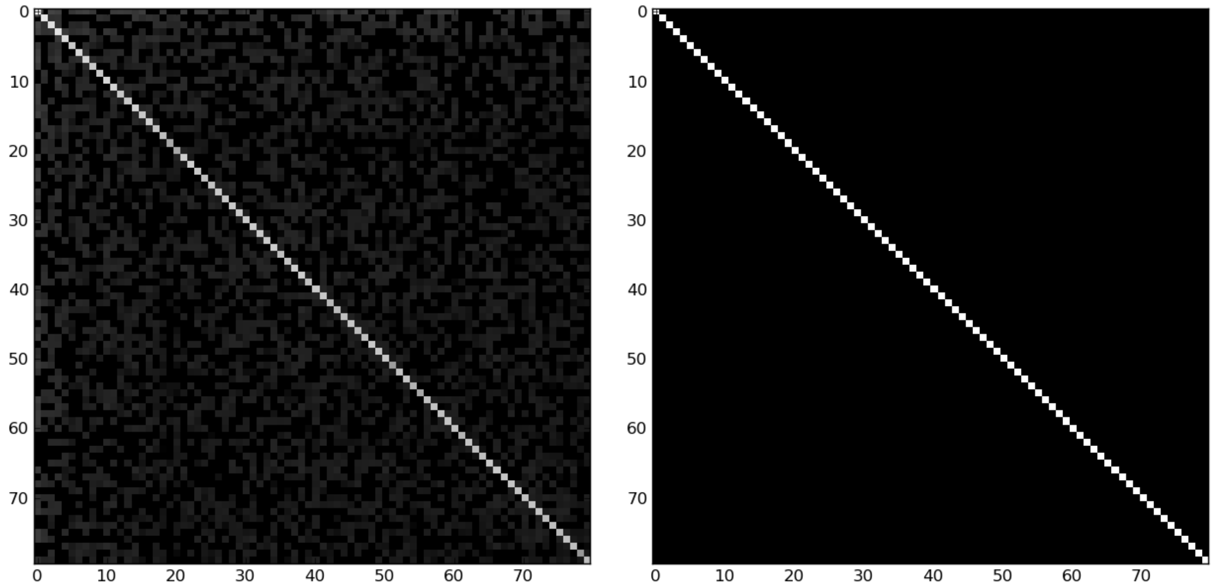
Gegeben ist ein Tensor dritter Stufe der Dimension (82,82,80), der 80 zweidimensionale Basisfunktionen der Größe 82x82px repräsentieren soll. Dies sind die ersten vier Basisbilder (vlnR):



Es wird nun behauptet, dass diese Bilder B^i eine Orthonormalbasis bilden, d.h. $\langle B^i, B^j\rangle=\delta_{ij}$.

Allerdings wurde in der Aufgabenstellung kein inneres Produkt definiert, und mir drängte sich zunächst keine eindeutige Definition auf. Daher nahm ich zunächst den mathematisch etwas unplausiblen Ansatz, ein inneres Produkt (*numpy.linalg.inner*, über die jeweils letzte Achse) zwischen allen B^i, B^j zu berechnen und dann eine Abbildung in den zugrundeliegenden Skalarkörper durch eine Summe über alle Elemente B^i_{lm} zu erzeugen. Das Ergebnis war eine Matrix $(M)^{ij}=\langle B^i, B^j\rangle=\sum_{lm}\sum_a B^i_{la}\cdot B^j_{ma}$. Ein sehr ähnliches Ergebnis bringt das regluäre Matrixprodukt $(M')^{ij}=\sum_{lm}\sum_a B^i_{la}\cdot B^j_{am}$. Der Orthonormalitätsbeweis wäre erbracht, wenn $M^{ij}=\delta_{ij}$ wenigstens qualitativ erfüllt wäre. Plottet man die Matrix als Bitmap, kann man erst mit einem komponentenweisen Logaritmus $(M_l)^{ij}=\log(M^{ij})$ den Hauch einer Orthogonalität feststellen.

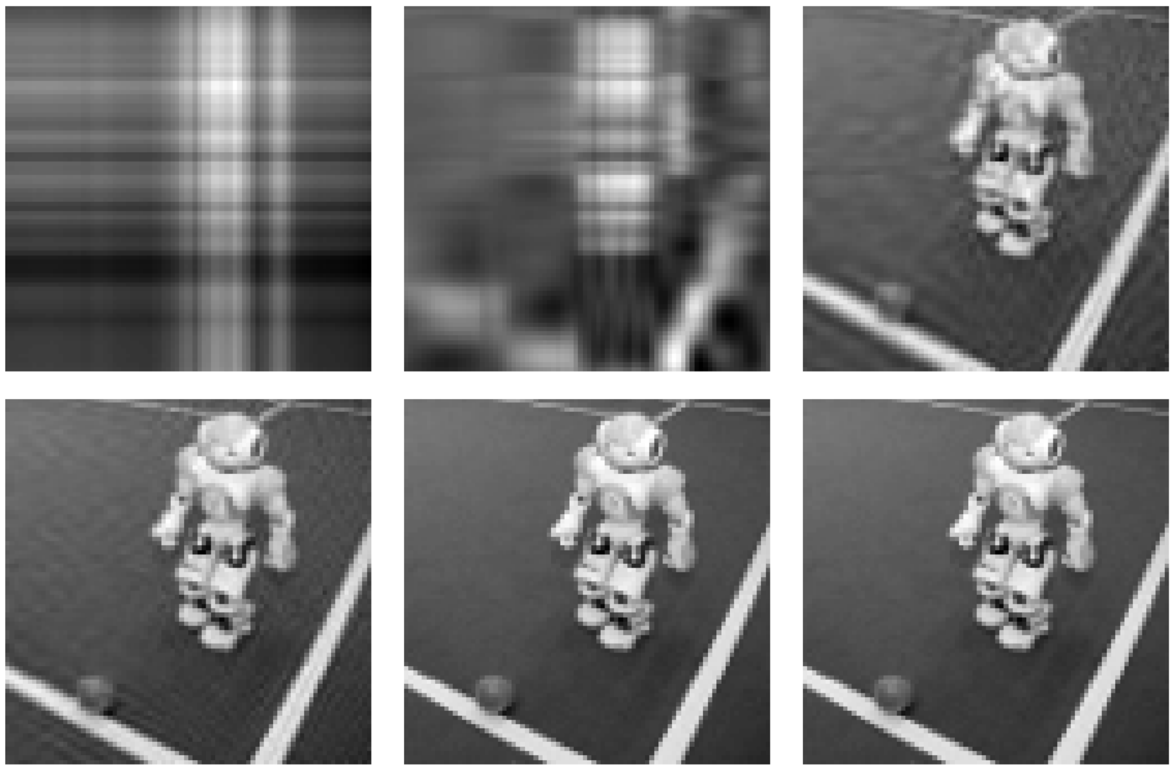
Dass die Aufgabe aber vorsah, die zweidimensionale Matrix als flaches eindimensionales Signal zu verstehen, d.h. die Matrix als Vektor zu interpretieren und darauf ein kanonisches Skalarprodukt $(M_f)^{ij}=\sum_{ab} B^i_a B^j_a$ zu definieren, wird erst klar, wenn man die Qualität der Ergebnisse vergleicht:



Links die Matrix M_i, rechts M_f. Die Einträge der Matritzen sind die jeweiligen Skalarprodukte $(M_k)^{ij}=\langle B^i, B^j \rangle_k$, man sieht links eine schwache, rechts eine starke Orthonormalität.

Hat man für einen Vektorraum erst ein Orthonormalsystem gefunden, lassen sich darin alle Vektoren besonders einfach ausdrücken:

$v=\sum_i a_i B^i$ wobei $a_i=\langle v, B\rangle$ die Entwicklungskoeffizienten sind. In diesem Fall liegen für einen Vektorraum, der anschaulich die Dimension 82*82=6724 hat, nur 80 Einheitsvektoren vor, sodass lediglich ein sehr kleiner Unterraum erreicht werden kann. Im Folgenden wird der bereits bekannte "nao-Bot" (von oben links bis unten rechts) mit den ersten 1, 4, 20, 30, allen 80 Basisfunktionen B^i entwickelt, das letzte Bild ist im Vergleich das Original:



Im vierten Bild wurden dabei mit dem flachen Skalarprodukt $\langle \cdots \rangle_f$ folgende vier Entwicklungskoeffizienten bestimmt:

$$a = \langle v, B \rangle_f = [\ 10550.87551779 \quad 1649.36720211 \quad 1343.16747956 \quad 952.92777363]$$

Das fünfte Bild ist nicht nur mit bloßem Auge vom Original zu unterscheiden, sondern es ist exakt das Original. Demnach ist die Orthonormalbasis vollständig für einen Unterraum V_u, der das Originalbild beinhaltet. Als allgemeiner Kompressionsalgorithmus für $dim(V_{82}) = 82 \cdot 82$ px-Bilder (80x64bit-Koeffizienten = 320bit Speichergröße statt 12kB JPG oder 1,7MB Rohbitmap) taugt dies Verfahren wegen der ungünstigen Abdeckung von gerade mal $dim(V_u) / dim(V_{82}) = 0{,}9\ \%$ aller Bilder nicht. (Soll heißen die orthonormalen Basisbilder sind gerade für dieses eine Bild gewählt)