

9.13 ΣΥΜΠΙΕΣΗ JPEG

Ο αλγόριθμος συμπίεσης JPEG ανήκει στην κατηγορία των τεχνικών συμπίεσης με απώλειες και βασίζεται στη χρήση του 2D-DCT. Ο όρος JPEG προέρχεται από τα αρχικά της ομάδας Joint Photographic Experts Group που ανέπτυξε και καθιέρωσε το πρότυπο αυτό διεθνώς το 1992. Έκτοτε, οι περισσότερες ψηφιακές εικόνες αποθηκεύονται και αποστέλλονται ως αρχεία jpg.

Η αποτελεσματικότητα της συμπίεσης κατά JPEG οφείλεται στις ακόλουθες τρεις παρατηρήσεις:

Παρατήρηση 1^η: Οι φωτεινότητες των εικόνων και το περιεχόμενό τους δεν μεταβάλλονται ευρέως και αρκετές φορές τοπικά. Δηλαδή, αν διασπάσουμε την εικόνα σε ένα άθροισμα μικρών ορθογώνιων περιοχών 8×8 , τότε περιμένουμε ότι στις περισσότερες από αυτές τις περιοχές θα έχουμε μικρή μεταβλητότητα στις φωτεινότητες και συνεπώς περίσσεια πληροφορίας.

Παρατήρηση 2^η: Ψυχοσωματικά πειράματα δείχνουν ότι οι άνθρωποι είναι πολύ λιγότερο πιθανό να αντιληφθούν την απώλεια πληροφορίας στις υψηλές συχνότητες παρά στις χαμηλές. Συνεπώς, η χωρική περίσσεια πληροφορίας μπορεί να μειωθεί σημαντικά με την κατάλληλη μείωση του υψίσυχνου περιεχομένου της εικόνας.

Παρατήρηση 3^η: Η οπτική οξύτητα είναι πολύ μεγαλύτερη σε γκρι αποχρώσεις απ' ότι σε έγχρωμες. Για το λόγο αυτό, οι χρωματικές συνιστώσες μπορούν να υποστούν άμεση υποδειγματοληψία πριν την κωδικοποίηση και τη συμπίεσή τους.

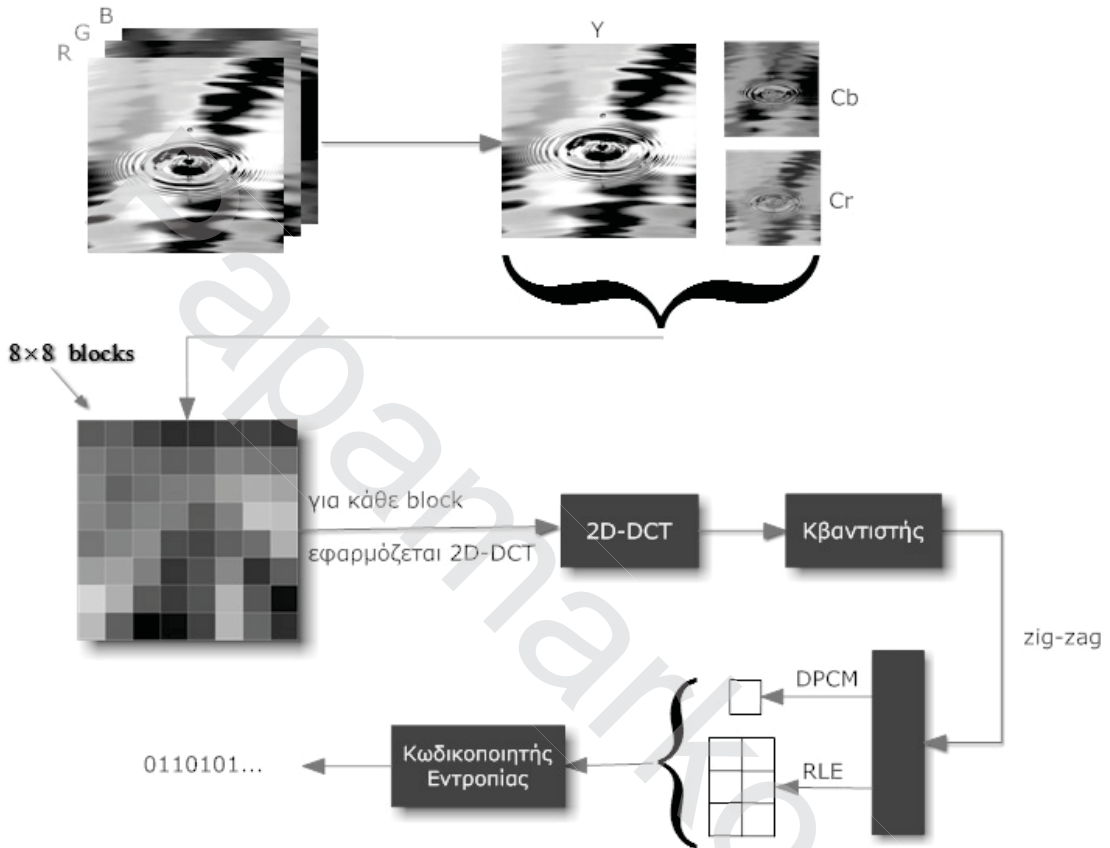
9.13.1 Κωδικοποίηση

Στο Σχήμα 9.23 παρουσιάζονται τα κύρια στάδια κωδικοποίησης που ακολουθεί ο αλγόριθμος JPEG. Τα στάδια αυτά αναλυτικά έχουν ως εξής

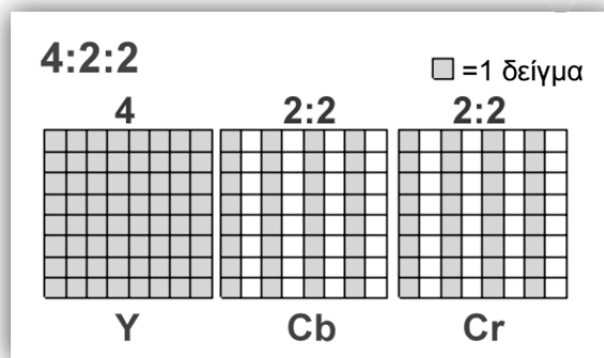
Στάδιο 1^ο: Διάσπαση της εικόνας

Στο πρώτο στάδιο, και εφόσον η εικόνα είναι έγχρωμη, διασπάται σύμφωνα με το χρωματικό μοντέλο YCbCr (ή σε YIQ), σε μια εικόνα φωτεινότητας Y καθώς και στις δύο πρόσθετες χρωματικές συνιστώσες Cb και Cr. Μια τέτοια διάσπαση είναι επιθυμητή, επειδή η πληροφορία που περιέχεται στη γκρι εικόνα συμβάλλει περισσότερο στην αντιληπτή ποιότητα της εικόνας από ότι οι πληροφορίες χρώματος. Αυτό οφείλεται στο γεγονός ότι το

ανθρώπινο μάτι χρησιμοποιεί κυρίως πληροφορίες από την κλίμακα του γκρι για τον εντοπισμό των ορίων. Έτσι, από οπτική άποψη, είναι αποδεκτό να απορρίψουμε περισσότερες πληροφορίες χρώματος απ’ ότι πληροφορίες που αφορούν αποχρώσεις του γκρι. Σύμφωνα με τα παραπάνω, οι δύο χρωματικές συνιστώσες υπόκεινται σε υποδειγματοληψία τύπου 4:2:2 σύμφωνα με τη διαδικασία που φαίνεται στο Σχήμα 9.24.



Σχήμα 9.22 Το γενικό σχήμα για τη συμπίεση κατά JPEG.



Σχήμα 9.24 Υποδειγματοληψία χρώματος 4:2:2.

Στάδιο 2^ο: Εφαρμογή του 2D-DCT

Στο δεύτερο στάδιο, η εικόνα Y , καθώς και οι δύο εικόνες που περιέχουν τις χρωματικές πληροφορίες, μετασχηματίζονται στο πεδίο των συχνοτήτων. Αυτή η διαδικασία σε κάθε μια από τις τρεις εικόνες αποτελείται από τη διαίρεση των εικόνων σε τετράγωνα (συνήθως 8×8) blocks και την εφαρμογή στη συνέχεια, σε κάθε ένα από αυτά, του 2D-DCT. Πριν τον μετασχηματισμό, για λόγους καλύτερης κβαντοποίησης και συμπίεσης, αφαιρείται από κάθε εικόνα η συνεχής της συνιστώσα $2^{n-1} = 128$ (n αριθμός των bits που συνήθως είναι ίσος με 8). Έστω λοιπόν $F_k(u, v)$, $(u, v) = 0, 1, \dots, 63$ οι συντελεστές του 2D-DCT που προκύπτουν για το block k . Όπως έχουμε ήδη αναφέρει σε προηγούμενο κεφάλαιο, ο 2D-DCT μας δίνει μια αποτελεσματική εκπροσώπηση στο χώρο των συχνοτήτων που ταιριάζει καλύτερα για τη συμπίεση. Συγκεκριμένα, με την εφαρμογή του 2D-DCT παράγονται συντελεστές το πλάτος των οποίων είναι μεγαλύτερο στις χαμηλές συχνότητες και για το λόγο αυτό οι συντελεστές στις χαμηλές συχνότητες είναι οι πλέον σημαντικοί. Αντίθετα, οι συντελεστές στις υψηλές συχνότητες έχουν συνήθως χαμηλό πλάτος και δεν είναι τόσο σημαντικοί.

Στάδιο 3^ο: Κβαντοποίηση

Στο τρίτο στάδιο, το κάθε μπλοκ των 2D-DCT συντελεστών υποβάλλεται σε μια διαδικασία κβαντοποίησης, η οποία έχει ως αποτέλεσμα την απώλεια πληροφορίας. Αυτό φυσικά αφορά τόσο τις γκρι αποχρώσεις όσο και τις πληροφορίες χρώματος. Η κβαντοποίηση γίνεται συνήθως με προκαθορισμένους πίνακες Q διαστάσεων 8×8 που καλούνται πίνακες κβαντισμού ενώ οι τιμές που περιέχουν ονομάζονται συντελεστές κβαντισμού. Συγκεκριμένα, κάθε συντελεστής του 2D-DCT διαιρείται με τον αντίστοιχο συντελεστή του πίνακα κβαντισμού και το αποτέλεσμα στρογγυλοποιείται. Η διαδικασία αυτή περιγράφεται από τη σχέση

$$\hat{F}_k(u, v) = \left\lfloor \frac{F_k(u, v)}{Q(u, v)} \right\rfloor, \quad (u, v) = 0, \dots, 7 \quad (9.40)$$

όπου $\hat{F}_k(u, v)$ οι νέοι συντελεστές του 2D-DCT.

Αρχικά, πίνακες κβαντισμού είχαν καθοριστεί από το πρότυπο JPEG τόσο για την αχρωματική όσο και για τις χρωματικές συνιστώσες και έχουν σχεδιαστεί σύμφωνα με ένα μοντέλο της ανθρώπινης αντίληψης. Ένα αξιοσημείωτο και εξαιρετικά χρήσιμο χαρακτηριστικό της διαδικασίας JPEG

είναι ότι σε αυτό το βήμα, διαφορετικά επίπεδα συμπίεσης και ποιότητας της εικόνας μπορούν να επιτευχθούν με την κατάλληλη επιλογή των πινάκων κβαντοποίησης. Αυτό επιτρέπει στο χρήστη να αποφασίσει για τα επίπεδα ποιότητας της τελικής συμπιεσμένης εικόνας που κυμαίνονται από 1 έως 100, όπου το 1 αντιστοιχεί στην χαμηλότερη ποιότητα και την υψηλότερη συμπίεση, ενώ το 100 δίνει την καλύτερη ποιότητα και το χαμηλότερο λόγο συμπίεσης. Ως αποτέλεσμα, μπορούμε να καθορίσουμε την ποιότητα και το βαθμό συμπίεσης ανάλογα με τις ανάγκες. Έχουν προταθεί διάφοροι πίνακες κβαντοποίησης από διαφορετικές εταιρείες (<http://www.impulseadventure.com/photo/jpeg-quantization.html>). Ορισμένοι, τυπικοί πίνακες κβαντισμού είναι οι ακόλουθοι, όπου ο δείκτης του Q καθορίζει το επίπεδο ποιότητας.

Πίνακες κβαντισμού για την Y συνιστούσα:

$$Q_{10} = \begin{bmatrix} 80 & 60 & 50 & 80 & 120 & 200 & 255 & 255 \\ 55 & 60 & 70 & 95 & 130 & 255 & 255 & 255 \\ 70 & 65 & 80 & 120 & 200 & 255 & 255 & 255 \\ 70 & 85 & 110 & 145 & 255 & 255 & 255 & 255 \\ 90 & 110 & 255 & 255 & 255 & 255 & 255 & 255 \\ 120 & 175 & 255 & 255 & 255 & 255 & 255 & 255 \\ 255 & 255 & 255 & 255 & 255 & 255 & 255 & 255 \\ 255 & 255 & 255 & 255 & 255 & 255 & 255 & 255 \end{bmatrix}$$

$$Q_{50} = \begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}$$

$$Q_{90} = \begin{bmatrix} 3 & 2 & 2 & 3 & 5 & 8 & 10 & 12 \\ 2 & 2 & 3 & 4 & 5 & 12 & 12 & 11 \\ 3 & 3 & 3 & 5 & 8 & 11 & 14 & 11 \\ 3 & 3 & 4 & 6 & 51 & 17 & 16 & 12 \\ 4 & 4 & 7 & 11 & 14 & 22 & 21 & 15 \\ 5 & 7 & 11 & 13 & 16 & 12 & 23 & 18 \\ 10 & 13 & 16 & 17 & 21 & 24 & 24 & 21 \\ 14 & 18 & 19 & 20 & 22 & 20 & 20 & 20 \end{bmatrix}$$

Πίνακας κβαντισμού για τις χρωματικές συνιστώσες:

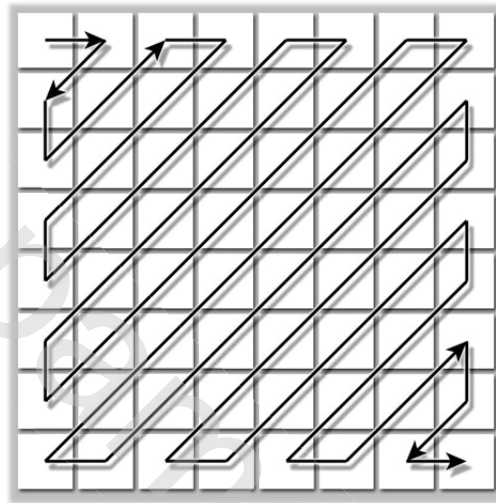
$$Q_{50} = \begin{bmatrix} 17 & 18 & 24 & 47 & 99 & 99 & 99 & 99 \\ 18 & 21 & 26 & 66 & 99 & 99 & 99 & 99 \\ 24 & 26 & 56 & 99 & 99 & 99 & 99 & 99 \\ 47 & 66 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \end{bmatrix}$$

Όπως μπορούμε να παρατηρήσουμε από τους παραπάνω πίνακες, οι μεγάλοι συντελεστές κβαντισμού για τους συντελεστές των υψηλών συχνοτήτων προκαλούν μηδενισμό των αντίστοιχων συντελεστών του 2D-DCT. Αυτό έχει ως αποτέλεσμα την εξάλειψη της επιρροής των λιγότερο σημαντικών πληροφοριών της εικόνας. Όμως, ο μηδενισμός των συντελεστών έχει ως αποτέλεσμα ο 8×8 πίνακας των συντελεστών του 2D DCT να περιέχει πολλά μηδενικά κάτι που εξασφαλίζει σημαντική συμπίεση των δεδομένων.

Στάδιο 4^ο: Zig-zag δειγματοληψία

Στο τέταρτο στάδιο, εφαρμόζεται μια διαδικασία δειγματοληψίας των συντελεστών του 2D-DCT τύπου zig-zag και κωδικοποίηση μήκους διαδρομής (run-length) σε κάθε μπλοκ των κβαντισμένων συντελεστών του 2D-DCT. Για το σκοπό αυτό χρησιμοποιείται μια διαδρομή zigzag (Σχήμα 9.25) η οποία καθορίζει τη σειρά των συντελεστών που θα κωδικοποιηθούν και με την οποία γίνεται εκμετάλλευση του πλήθους των διαδοχικών

μηδενικών που εμφανίζονται σε κάθε μπλοκ. Η διαδρομή zigzag προχωρεί από χαμηλής συχνότητας σε υψηλής συχνότητας συντελεστές και συνεπώς λόγω της κβαντοποίησης και της ύπαρξης συντελεστών υψηλής συχνότητας η διαδρομή κάθε κωδικοποιημένου μπλοκ θα περιέχει συνήθως τουλάχιστον ένα μεγάλο αριθμό από συνεχόμενα μηδενικά στο τέλος. Έτσι, το μέγεθος του χώρου που απαιτείται για την αντιπροσώπευση κάθε μπλοκ μπορεί να μειωθεί ουσιαστικά με την αντικατάσταση των συνεχόμενων μηδενικών με το συμβολισμό $(0, n)$, όπου n το πλήθος των μηδενικών.



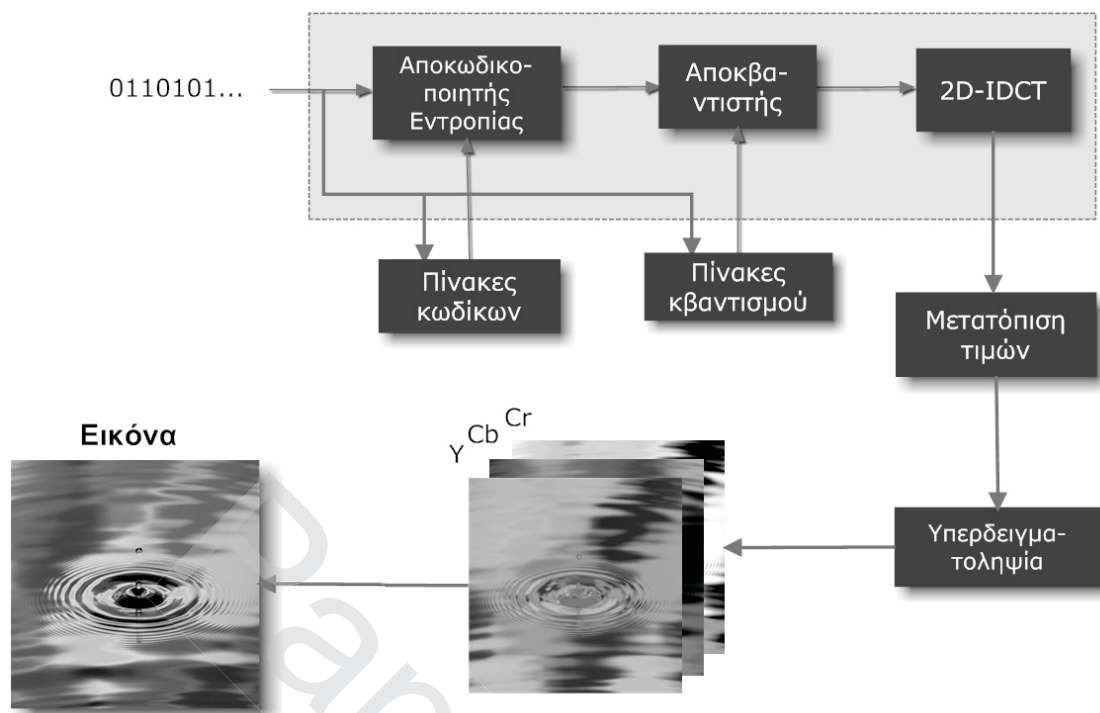
Σχήμα 9.25 Διαδρομή zig-zag

Στάδιο 5^ο: Κωδικοποίηση και συμπίεση

Στο πέμπτο και τελευταίο στάδιο, τα δεδομένα που προκύπτουν μπορεί να συμπιεστούν περαιτέρω με μια χωρία απώλειες κωδικοποίηση όπως είναι η Huffman ή η αριθμητική κωδικοποίηση. Σημειώνεται ότι οι DC συντελεστές $(0,0)$ μπορούν να συμπιεστούν με τη βοήθεια DPCM ως η διαφορά των DC τιμών δύο διαδοχικών blocks. Τα συμπιεσμένα δεδομένα που προκύπτουν μπορούν στη συνέχεια να αποθηκευθούν σε ένα μέσο αποθήκευσης και να αναπαραχθούν με μια αντίστροφη διαδικασία αποσυμπίεσης/αποκωδικοποίησης.

9.13.2 Αποκωδικοποίηση

Κατά την διαδικασία αποκωδικοποίησης όλα τα παραπάνω στάδια εκτελούνται αντίστροφα. Η όλη διαδικασία αντιστροφής φαίνεται περιληπτικά στο Σχήμα 9.26 Συγκεκριμένα έχουμε τα ακόλουθα στάδια:



Σχήμα 9.26 Διαδικασία JPEG αποκωδικοποίησης.

Στάδιο 1^ο: Αποκωδικοποίηση

Αρχικά, για κάθε επιμέρους 8×8 block, γίνεται αποκωδικοποίηση των συντελεστών και λαμβάνοντας υπόψη ότι χρησιμοποιήθηκε διαδρομή zig-zag, ανακτώνται οι συντελεστές $\hat{F}_k(u, v)$ του 2D-DCT.

Στάδιο 2^ο: Αποκβαντοποίηση

Τους συντελεστές $\hat{F}_k(u, v)$ τους πολλαπλασιάζουμε με τους πίνακες κβαντισμού και έτσι ανακτούμε τους συντελεστές του 2D-DCT. Δηλαδή

$$F_k(u, v) = \hat{F}_k(u, v)Q(u, v) \quad (9.41)$$

Στάδιο 3^ο: Εφαρμογή 2D-IDCT

Σε κάθε επιμέρους block εφαρμόζουμε αρχικά τον 2D-IDCT

$$\hat{f}_k(m, n) = 2D-IDCT[F_k(u, v)], \quad (m, n) = 0, \dots, 7 \quad (9.42)$$

Στη συνέχεια, στις τιμές $\hat{f}_k(m, n)$ προστίθεται η μετατόπιση $2^{n-1} = 128$, οπότε

$$f_k(m, n) = \hat{f}_k(m, n) + 128, \quad (m, n) = 0, \dots, 7 \quad (9.43)$$

Στάδιο 4^ο: Υπερδειγματοληψία στις χρωματικές συνιστώσες

Προκειμένου οι χρωματικές συνιστώσες Cb και Cr να αποκτήσουν τις αρχικές διαστάσεις πρέπει να κάνουμε υπερδειγματοληψία στους

αντίστοιχους πίνακες $f_k(m,n)$. Η υπερδειγματοληψία μπορεί να γίνει με διάφορους τρόπους με απλούστερο αυτό της παρεμβολής.

Στάδιο 5^ο: Συνένωση των επιμέρους blocks και ανακατασκευή της αρχικής εικόνας

Στο τελικό στάδιο γίνεται αρχικά κατάλληλη συνένωση των επιμέρους block και σχηματίζονται οι εικόνες Y, Cb, Cr και στη συνέχεια η τελική έγχρωμη εικόνα.

Πρέπει να σημειώσουμε ότι η συμπίεση με JPEG εκτός του ότι είναι μια συμπίεση με απώλειες, προξενεί και παραμορφώσεις στην εικόνα. Για παράδειγμα, ο κατακερματισμός της εικόνας σε επιμέρους blocks δημιουργεί κατά την ανασύνθεσή της προβλήματα στα όρια μεταξύ των blocks. Για την αντιμετώπιση των προβλημάτων αυτών έχουν προταθεί διάφορες τεχνικές μεταξύ των οποίων είναι αυτές που βασίζονται σε εξομάλυνση των ορίων με παράλληλη διατήρηση των ακμών. Όμως, προκειμένου να εξασφαλίζεται οι διαδικασίες συμπίεσης/αποσυμπίεσης του προτύπου, οι τεχνικές αυτές αφορούν μετα-επεξεργασία των εικόνων, δηλαδή εφαρμόζονται μετά τα αποτελέσματα του JPEG.

Παράδειγμα 9.12 Χωρίς να κάνετε κωδικοποίηση εφαρμόστε τον JPEG αλγόριθμο στην εικόνα του Σχήματος 9.27. Η εικόνα είναι γκρι αποχρώσεων και διαστάσεων 256×256 pixels. Για την κβάντιση των συντελεστών χρησιμοποιείστε τους πίνακες κβαντισμού Q_{10} και Q_{50} . Και για τις δύο περιπτώσεις προσδιορίστε τις συμπιεσμένες εικόνες που προκύπτουν καθώς και τις τιμές PSNR.



Σχήμα 9.27 Αρχική εικόνα.

Λύση

Η εικόνα αρχικά διαχωρίζεται σε $32 \times 32 = 1024$ blocks διαστάσεων 8×8 pixels το καθένα. Για παράδειγμα, το block 7×10 έχει τιμές

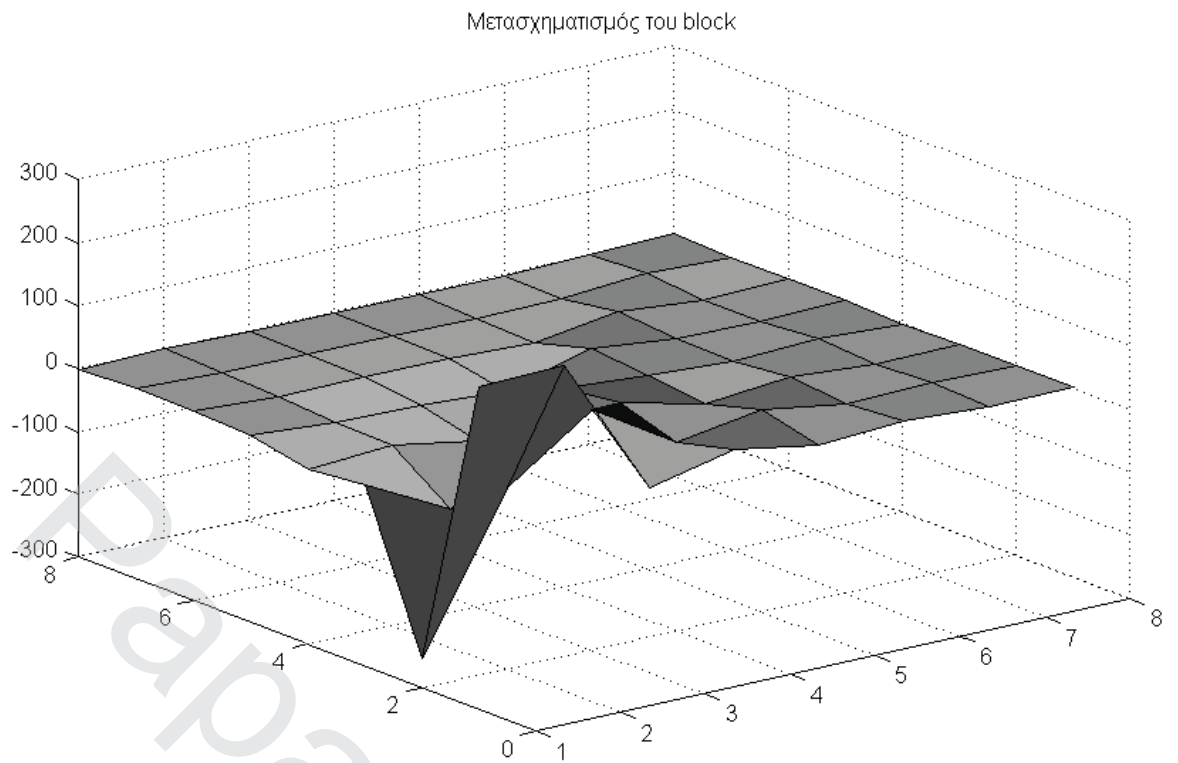
$$\begin{bmatrix} 140 & 108 & 97 & 100 & 98 & 100 & 100 & 100 \\ 184 & 145 & 111 & 100 & 98 & 100 & 98 & 100 \\ 199 & 185 & 159 & 122 & 106 & 102 & 102 & 101 \\ 200 & 194 & 185 & 169 & 134 & 106 & 106 & 105 \\ 201 & 200 & 200 & 189 & 171 & 132 & 102 & 101 \\ 209 & 208 & 206 & 198 & 188 & 175 & 132 & 104 \\ 213 & 213 & 210 & 205 & 196 & 191 & 171 & 122 \\ 213 & 215 & 215 & 212 & 205 & 195 & 184 & 157 \end{bmatrix}$$

Αφαιρούμε στη συνέχεια το 128 από όλες τις τιμές και κατόπιν εφαρμόζουμε 2D-DCT σε κάθε επιμέρους block. Για το παραπάνω block προκύπτουν οι ακόλουθες τιμές των συντελεστών του 2D-DCT

$$\begin{bmatrix} 211.8750 & 216.6249 & -8.2801 & 23.4039 & -0.8750 & 7.9719 & 0.5884 & 0.9029 \\ -253.6712 & -11.8411 & 80.9133 & -1.0200 & 20.6708 & -1.0894 & 4.5205 & -0.6240 \\ -11.7843 & -80.8798 & 15.6935 & 28.0254 & -4.1323 & 9.0930 & -1.8562 & -0.6045 \\ -21.3349 & -12.7267 & -37.8277 & 15.1416 & 6.1622 & -1.2344 & 5.6130 & -2.9170 \\ -2.3750 & -13.0848 & -14.4863 & -22.4252 & 14.8750 & 0.7043 & -1.9822 & 1.6853 \\ 1.3526 & -4.7347 & -4.0316 & -5.3380 & -9.4965 & 9.4754 & -4.2846 & 0.2283 \\ 2.5811 & -0.8094 & -1.8562 & -3.4352 & -5.7298 & -4.1652 & 4.5565 & -0.2271 \\ -2.2599 & 2.7724 & -1.7634 & -1.8656 & -3.4644 & -3.3684 & 0.7506 & 3.2240 \end{bmatrix}$$

Οι τιμές αυτές του μετασχηματισμού φαίνονται παραστατικά στο Σχήμα 9.28.

Η κβάντιση των συντελεστών με τον Q_{10} (διαίρεση και στρογγυλοποίηση στον κοντινότερο ακέραιο) μας δίνει τις ακόλουθες τιμές



Σχήμα 9.28 2D-DCT του block 7×10 .

$$\begin{bmatrix} 3 & 4 & 0 & 0 & 0 & 0 & 0 & 0 \\ -5 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Αυτό φυσικά γίνεται σε όλα τα blocks και οι συντελεστές αυτοί είναι που στη συνέχεια κωδικοποιούνται.

Για την αποκωδικοποίηση/απόσυμπίεση της εικόνας πρέπει αρχικά να πολλαπλασιάσουμε τις κβαντισμένες τιμές με τις αντίστοιχες τιμές του πίνακα Q_{10} . Θα πάρουμε δηλαδή τις ακόλουθες αποκβαντισμένες τιμές

$$\begin{bmatrix} 240 & 240 & 0 & 0 & 0 & 0 & 0 & 0 \\ -275 & 0 & 70 & 0 & 0 & 0 & 0 & 0 \\ 0 & -65 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Στη συνέχεια, εφαρμόζουμε σε κάθε block 2D-IDCT και στις τιμές που παίρνουμε προσθέτουμε το 128 και στη συνέχεια στρογγυλοποιούμε σε ακέραιες τιμές. Για το block 7×10 προκύπτει οι τιμές

$$\begin{bmatrix} 153 & 140 & 119 & 100 & 89 & 89 & 94 & 99 \\ 167 & 153 & 132 & 111 & 97 & 92 & 93 & 96 \\ 188 & 175 & 154 & 131 & 113 & 100 & 94 & 92 \\ 208 & 198 & 179 & 157 & 134 & 115 & 102 & 95 \\ 221 & 214 & 201 & 182 & 159 & 137 & 118 & 108 \\ 224 & 222 & 216 & 203 & 185 & 162 & 141 & 128 \\ 220 & 223 & 224 & 219 & 205 & 184 & 163 & 149 \\ 217 & 222 & 227 & 227 & 216 & 197 & 176 & 163 \end{bmatrix}$$

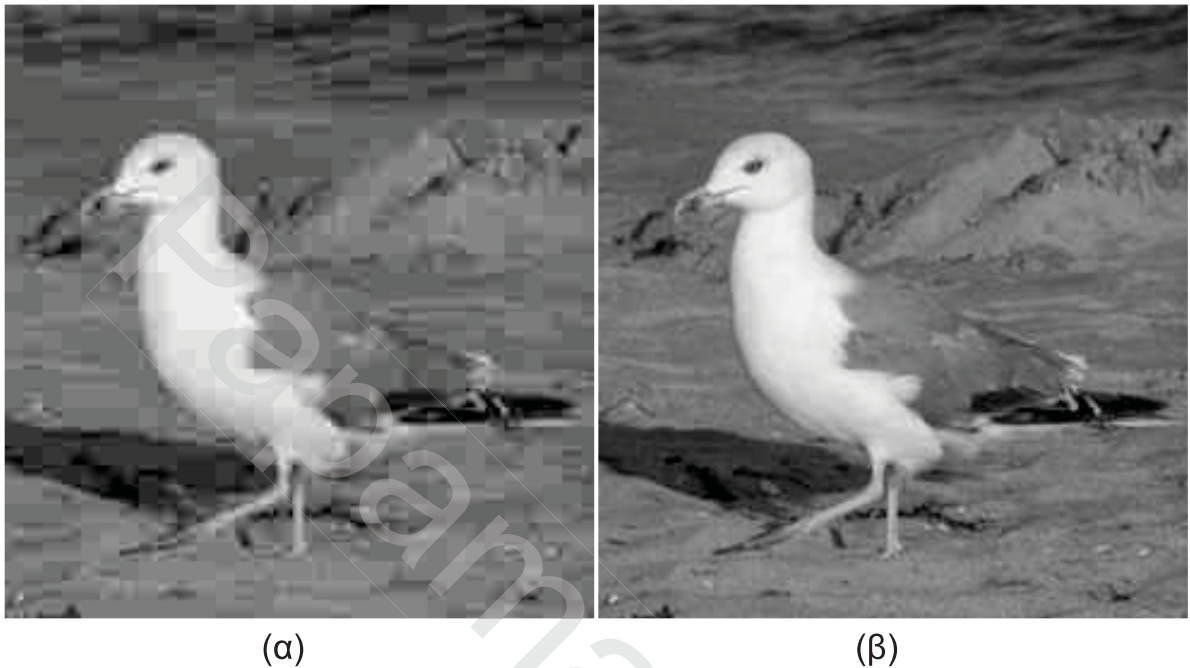
Αν αντί για τον Q_{10} χρησιμοποιήσουμε τον Q_{50} οι τιμές για το συγκεκριμένο block προκύπτουν ίσες με

$$\begin{bmatrix} 143 & 110 & 86 & 90 & 102 & 103 & 99 & 98 \\ 175 & 148 & 121 & 107 & 98 & 91 & 93 & 103 \\ 202 & 185 & 161 & 134 & 106 & 89 & 95 & 109 \\ 204 & 198 & 185 & 161 & 132 & 111 & 104 & 107 \\ 198 & 199 & 196 & 185 & 167 & 142 & 115 & 96 \\ 202 & 206 & 206 & 201 & 191 & 168 & 129 & 95 \\ 209 & 215 & 215 & 207 & 200 & 186 & 156 & 125 \\ 211 & 219 & 218 & 207 & 201 & 197 & 181 & 161 \end{bmatrix}$$

Μπορούμε να παρατηρήσουμε ότι στην δεύτερη περίπτωση, οι τιμές φωτεινότητας, όπως αναμέναμε, βρίσκονται κοντύτερα προς τις αρχικές. Η διαφορά αυτή στη ποιότητα φαίνεται και οπτικά από τη σύγκριση των

εικόνων που παίρνουμε (Σχήμα 9.29) αλλά και από τις τιμές PSNR για τις οποίες προκύπτουν οι ακόλουθες τιμές

$$PSNR_{Q_{10}} = 30.3785 \quad \text{και} \quad PSNR_{Q_{50}} = 35.3396$$



Σχήμα 9.29 (α), (β) Συμπίεση εικόνας με Q_{10} και Q_{50} , αντίστοιχα.

Το πρόγραμμα σε κώδικα Matlab του παραδείγματος δίδεται στο Σχήμα 9.30

```
% (c) 2011 Nikos Papamarkos
clear all;close all; clc;
I=imread('pouli256.bmp');
I=double(I);
[mI,nI]=size(I);
mb=mI/8; nb=nI/8;
figure (1)
imshow(uint8(I))
% Demo block
m=7;n=10;
A=I(1+8*m:8+8*m,1+8*n:8+8*n);
disp(' Ένα block της εικόνας')
disp(A)
% Αφαίρεση του 128
Is=I-128;
A=Is(1+8*m:8+8*m,1+8*n:8+8*n);
```

```

disp(' Μετά την αφαίρεση του 128')
disp(A)
% Εφαρμογή του 2D-DCT σε κάθε block 8x8
fun = @dct2;
J = blkproc(Is, [8 8], fun);
figure (2)
imagesc(J), colormap(copper)
title(' 2D-DCT ')
figure (3)
surf(J);colormap(Cool)
title(' 2D-DCT ')
B=J(1+8*m:8+8*m,1+8*n:8+8*n);
disp(' Μετασχηματισμός του block ')
disp(B)
figure (4)
surf(B);colormap(HSV)
title(' Μετασχηματισμός του block ')
% Πίνακας κβαντισμού
% Q = [16 11 10 16 24 40 51 61
%      12 12 14 19 26 58 60 55
%      14 13 16 24 40 57 69 56
%      14 17 22 29 51 87 80 62
%      18 22 37 56 68 109 103 77
%      24 35 55 64 81 104 113 92
%      49 64 78 87 103 121 120 101
%      72 92 95 98 112 100 103 99];
Q = [80 60 50 80 120 200 255 255
     55 60 70 95 130 255 255 255
     70 65 80 120 200 255 255 255
     70 85 110 145 255 255 255 255
     90 110 255 255 255 255 255 255
     120 175 255 255 255 255 255 255
     255 255 255 255 255 255 255 255
     255 255 255 255 255 255 255 255];
% Κβαντισμός των συντελεστών
Jquant=round(blkproc(J, [8 8], 'divq', Q));
C=Jquant(1+8*m:8+8*m,1+8*n:8+8*n);
disp(' Κβαντισμένοι συντελεστές')
disp(C)
% Αντιστροφή
% Εφαρμογή του 2D-IDCT σε κάθε block 8x8
% fun2 = @idct2;
Jdquant=blkproc(Jquant, [8 8], 'multip', Q);
disp(' Αποκβαντισμένες τιμές του block')
Jdquant(1+8*m:8+8*m,1+8*n:8+8*n)
Iinv = blkproc(Jdquant, [8 8], 'idct2');
disp(' Αντίστροφες τιμές του block')
Iinv(1+8*m:8+8*m,1+8*n:8+8*n)
Iinv=Iinv+128;
Im=round(Iinv);
disp(' Αντίστροφες τιμές του block με 128')
round(Iinv(1+8*m:8+8*m,1+8*n:8+8*n))
figure (6)
imshow(uint8(Im))
imwrite(uint8(Im), 'Comimage.bmp', 'BMP');
psnr(I, Im);
function p = psnr(x, y, vmax)
if nargin<3

```

```
m1 = max( abs(x(:)) );
m2 = max( abs(y(:)) );
vmax = max(m1,m2);
end
d = mean( (x(:)-y(:)).^2 );
p = 10*log10( vmax^2/d );
disp(' PSNR')
disp(p)
function y=divq(x,q)
y =x./q;
function y=multip(x,q)
y =x.*q;
```

Σχήμα 9.30 Κώδικας JPEG για το Παράδειγμα 9.12.

9.14 ΤΟ ΠΡΟΤΥΠΟ JPEG 2000

Το πρότυπο JPEG χρησιμοποιείται εδώ και μια δεκαετία περίπου. Έχει αποδειχτεί ένα πολύτιμο εργαλείο κατά την διάρκεια αυτών των χρόνων, αλλά δεν μπορεί να καλύψει τις υψηλού επιπέδου απαιτήσεις του σήμερα. Η ακίνητη εικόνα σήμερα είναι πολύ απαιτητική, όχι μόνο από την πλευρά της ποιότητας, αλλά και από την άποψη του μεγέθους. Το μέγεθος μιας εικόνας καλύπτει τάξεις μεγέθους, κυμαινόμενο από λογότυπα με μέγεθος μικρότερο από 100 Kbits μέχρι υψηλής ποιότητας σαρωμένες εικόνες μεγέθους 40 Gbits. Η ψηφιακή εικόνα έχει γίνει ένα αναπόσπαστο κομμάτι της ζωής μας, και η ανάγκη για ένα νέο ισχυρό εργαλείο που θα προσφέρει υψηλού επιπέδου δυνατότητες σε σχεδιαστές και χρήστες εφαρμογών εικόνας ήταν έντονη.

Η ιστορία ανάπτυξης του πρότυπου JPEG 2000 έχει περιληπτικά ως εξής. Στην αρχή της δεκαετίας του 1990, προτάθηκε ένας σημαντικός αριθμός νέων τεχνικών συμπίεσης, όπως για παράδειγμα οι τεχνικές CREW (Compression with Reversible Embedded w]Wavelets) [ZASB95] και EZW (Embedded Zerotree Wavelet) [SHAP93]. Οι τεχνικές αυτές είχαν ως στόχο όχι μόνον την καλύτερη συμπίεση, αλλά επίσης νέα χαρακτηριστικά στη συμπίεση που δεν υπήρχαν προηγουμένως. Με φόντο τα δεδομένα αυτά, τον Μάρτιο του 1997 έλαβε χώρα μια παγκόσμια πρόσκληση για τη συμμετοχή στην ανάπτυξη ενός καινούριου προτύπου για τη συμπίεση ακίνητης εικόνας, το πρότυπο JPEG 2000. Η πρώτη αξιολόγηση έγινε τον Νοέμβριο του 1997 στο Sydney της Αυστραλίας, όπου 24 αλγόριθμοι υποβλήθηκαν και αξιολογήθηκαν. Με βάση αυτήν την