

Βασικές αρχές παρεμβολής σημάτων και εικόνων

Κ. Δελήμπασης

ΠΜΣ : Υπολογιστική Ιατρική και Βιολογία

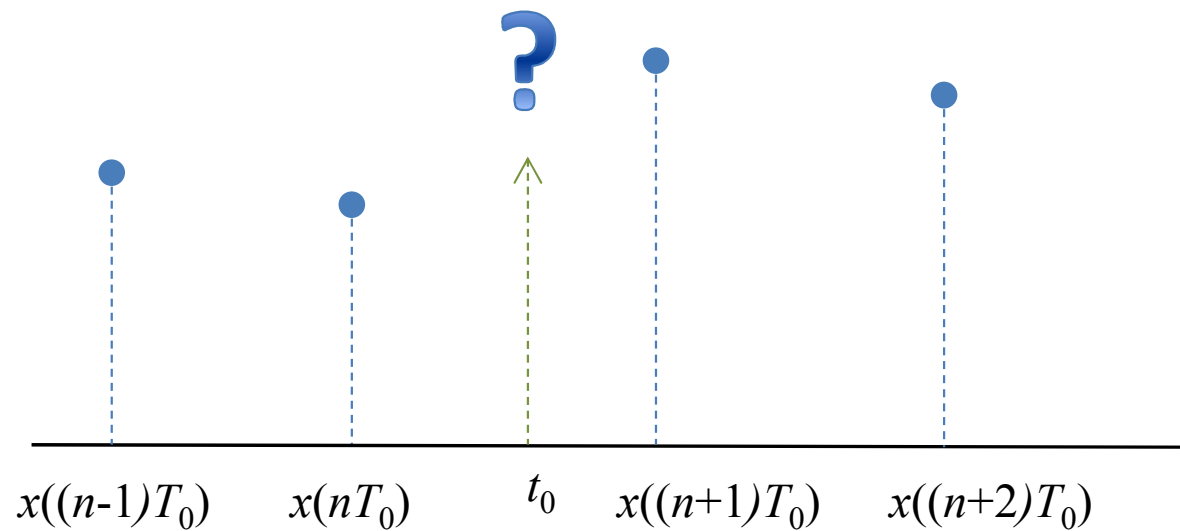
Μάθημα: Ειδικά θέματα Αριθμητικής Ανάλυσης
και Εφαρμοσμένων Μαθηματικών

Παρεμβολή (interpolation)

- Τα διακριτά σήματα προκύπτουν από δειγματοληψία στο χρόνο (σήμα 1D) ή στο χώρο (εικόνα 2 ή περισσότερων διαστάσεων)
- Συχνά χρειάζεται η τιμή του σήματος σε μία χρονική στιγμή μεταξύ δύο δειγμάτων
- Αυτό το πρόβλημα το αντιμετωπίζει η **παρεμβολή**

Παρεμβολή (interpolation)

- Τα διακριτά σήματα προκύπτουν από δειγματοληψία στο χρόνο (σήμα 1D) ή στο χώρο (εικόνα 2 ή περισσότερων διαστάσεων)
- Συχνά χρειάζεται η τιμή του σήματος σε μία χρονική στιγμή μεταξύ δύο δειγμάτων
- Αυτό το πρόβλημα το αντιμετωπίζει η **παρεμβολή**



Τα δείγματα ενός διακριτού σήματος και η ζητούμενη τιμή του σήματος σε μία ενδιάμεση θέση.

Παρεμβολή (interpolation)

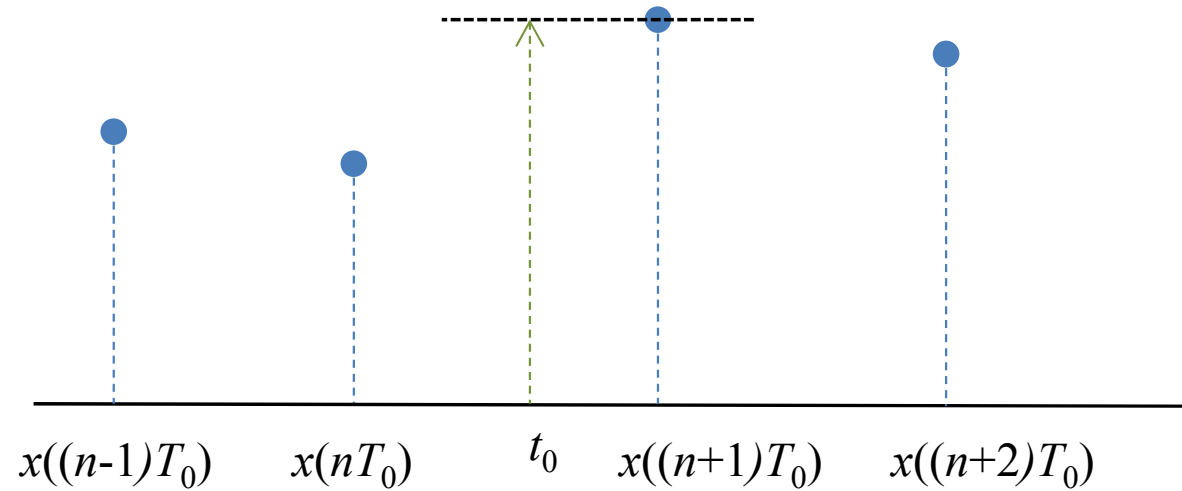
- **Ορισμός της παρεμβολής:**
 - Δεδομένης μίας σειράς μετρήσεων μίας ποσότητας για συγκεκριμένες τιμές της ανεξάρτητης μεταβλητής, ζητείται βρεθεί μία συνάρτηση, η οποία να έχει τιμή ίση με τις μετρήσεις στις θέσεις της ανεξάρτητης μεταβλητής.
- Η συνάρτηση μπορεί να χρησιμοποιηθεί για τον υπολογισμό της ποσότητας αυτής σε άλλες θέσεις στις οποίες δεν υπάρχει μέτρηση

Εφαρμογές

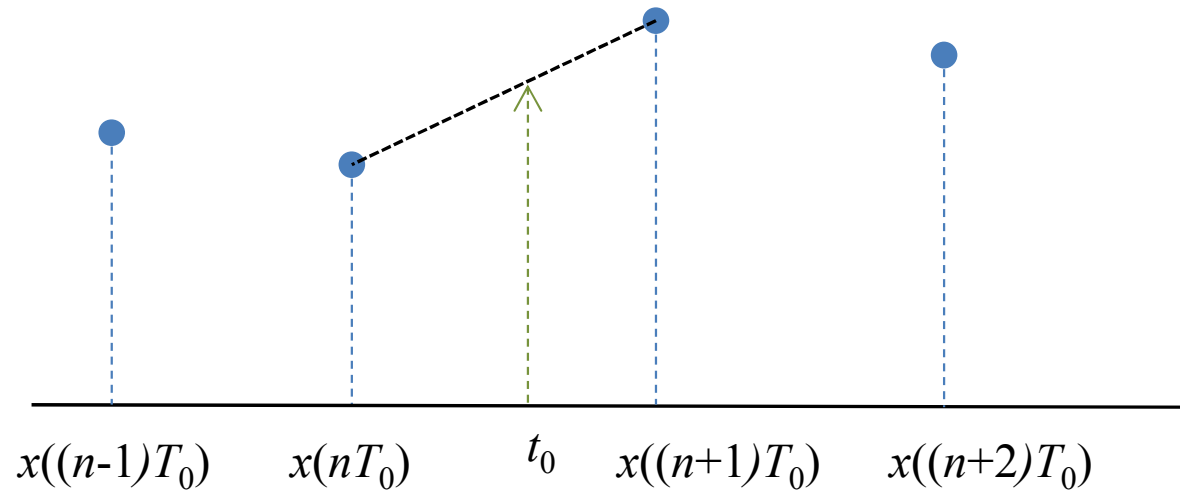
- Παρεμβολή διακριτών σημάτων
- Παρεμβολή σε περισσότερες διαστάσεις:
 - 2D εικόνες (απαιτείται σε οποιαδήποτε περιστροφή / αλλαγή κλίμακας κλπ).
 - 1D video frame (προσθήκη frame)
 - 3D volumes (απαιτείται σε οποιαδήποτε περιστροφή / αλλαγή κλίμακας κλπ)

Τα απλούστερα είδη παρεμβολής

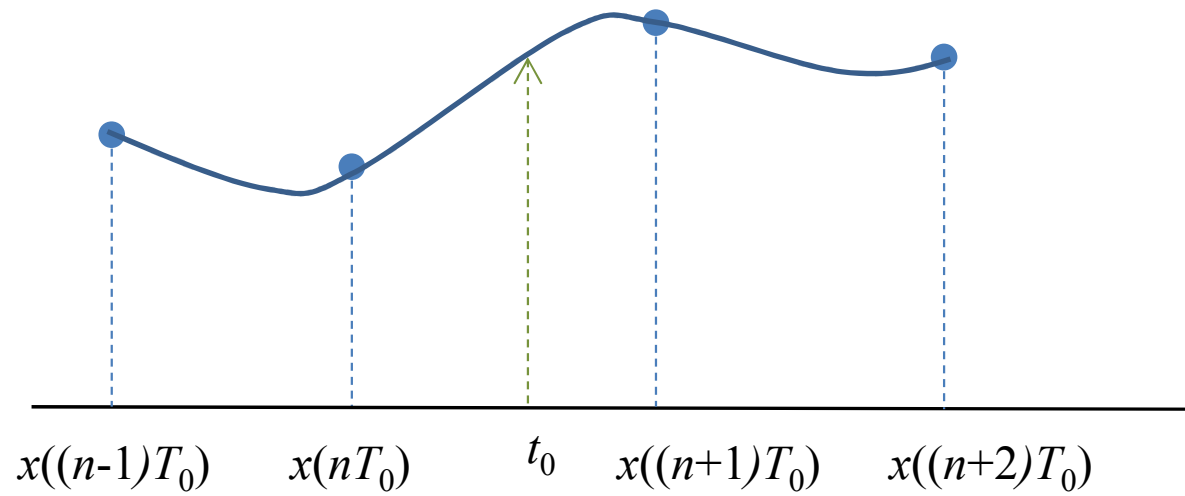
- Παρεμβολή κοντινότερου γείτονα
- Γραμμική παρεμβολή
- Παρεμβολή σε περισσότερες διαστάσεις:
 - 2D εικόνες
 - 1D video frame
 - 3D volumes



Παρεμβολή του σήματος στο $t=t_0$ με χρήση κοντινότερου γείτονα



Γραμμική παρεμβολή του σήματος στο $t=t_0$



Παρεμβολή του σήματος στο $t=t_0$ με κυβικό πολυώνυμο

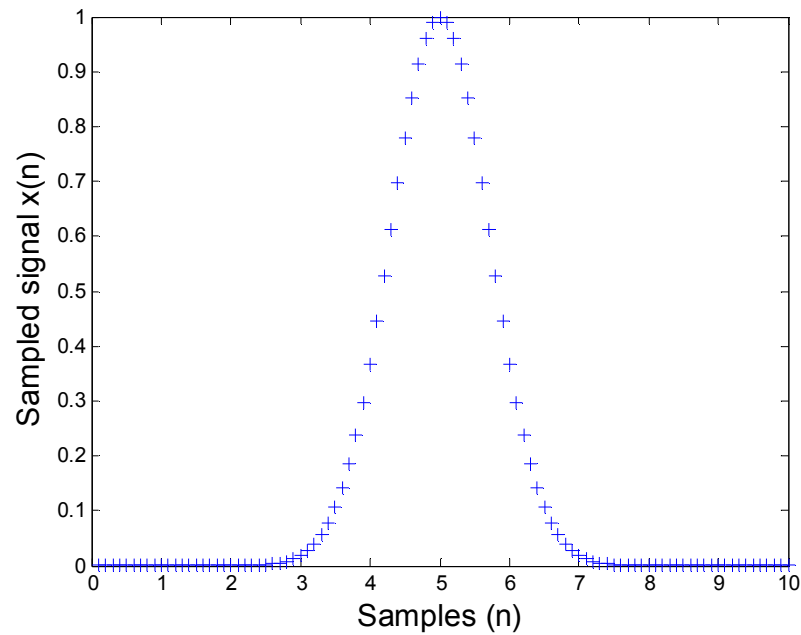
Ιδανική παρεμβολή

The ideal interpolator

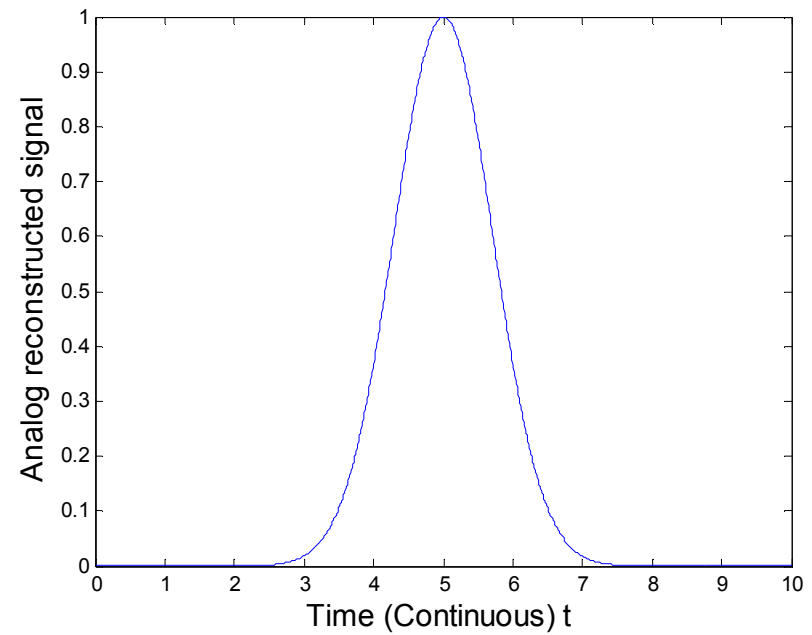
- Δεδομένου ενός διακριτού σήματος, συχνά απαιτείται η διαδικασία ανακατασκευής του αναλογικού σήματος Digital to Analog Conversion –DAC, δηλ η εύρεση της τιμής του σήματος σε οποιαδήποτε χρονική στιγμή t .
- Έστω $x(n)$ δειγματοληπτημένο σήμα από το οποίο θέλουμε να ανακατασκευάσουμε το αναλογικό $x(t)$. Τότε:

$$x(t) = \sum_{n=-\infty}^{\infty} x(n) \frac{\sin\left(\frac{\pi(t-nT_s)}{T_s}\right)}{\pi(t-nT_s)/T_s} = \sum_{n=-\infty}^{\infty} x(n) \operatorname{sinc}\left(\frac{(t-nT_s)}{T_s}\right)$$

- Η προηγούμενη σχέση δε μπορεί να υλοποιηθεί υπολογιστικά διότι:
 - Ο πυρήνας (kernel) $\text{sinc}(t)$ εκτείνεται από $t \rightarrow -\infty$ έως $t \rightarrow +\infty$ (μη αιτιατός πυρήνας)
- Παρ' όλα αυτά τη μελετάμε για λόγους πληρότητας
- Ακολουθεί παράδειγμα εφαρμογής της παρεμβολής με

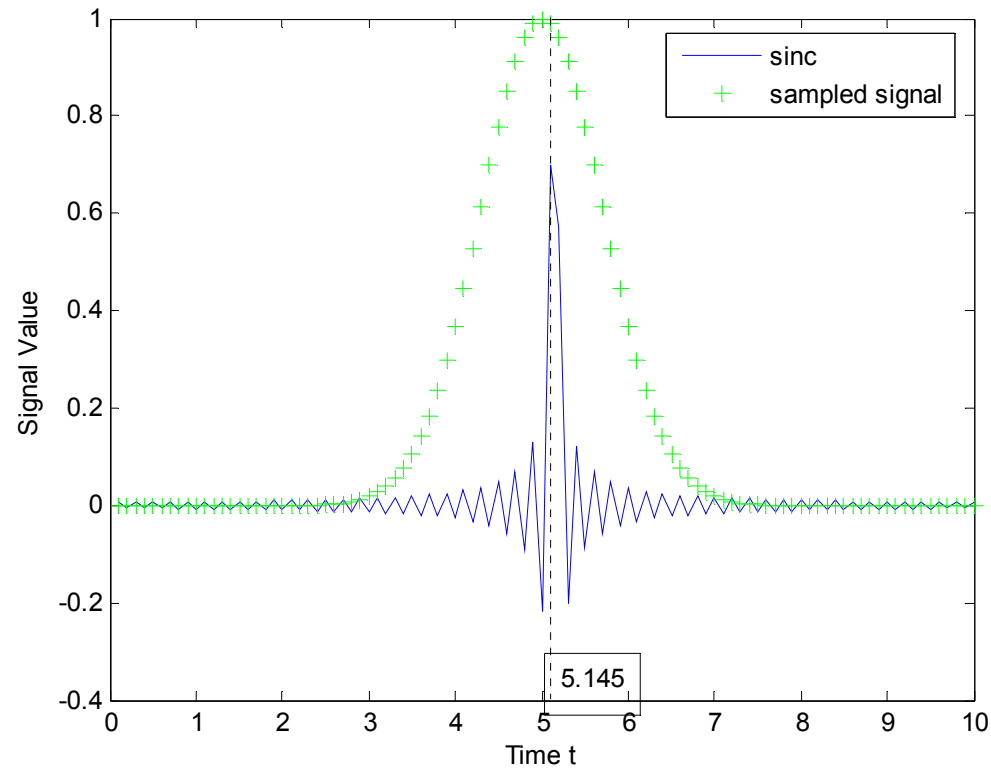


Δειγματοληπτημένο σήμα



Συνεχές σήμα

Παράδειγμα ανακατασκευής DAC μίας δειγματοληπτημένης γκαουσιανής

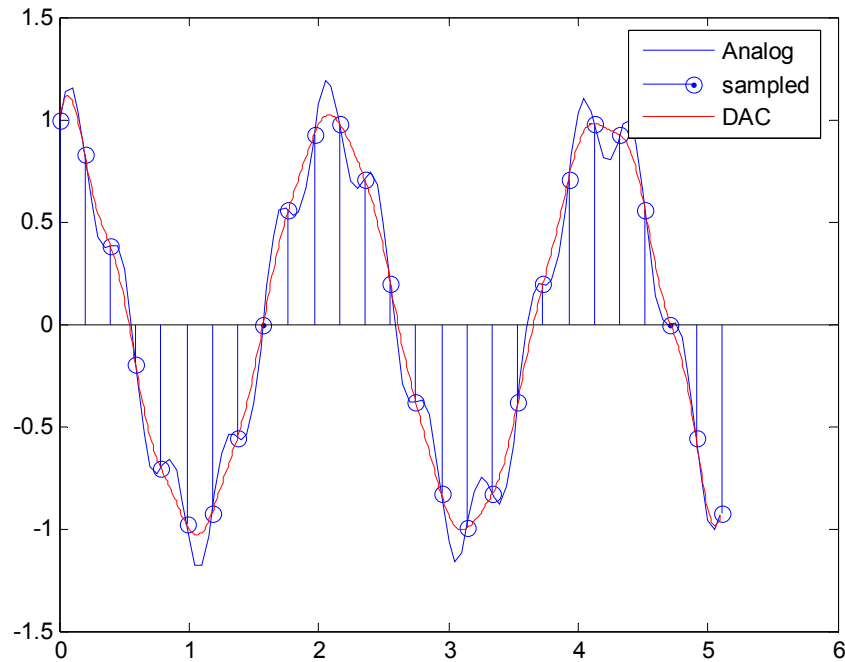


- Επεξήγηση του τύπου ανακατασκευής DAC
- Μπλε καμπύλη: ο μεταωπισμένος πυρήνας $\text{sinc}(t_0)$

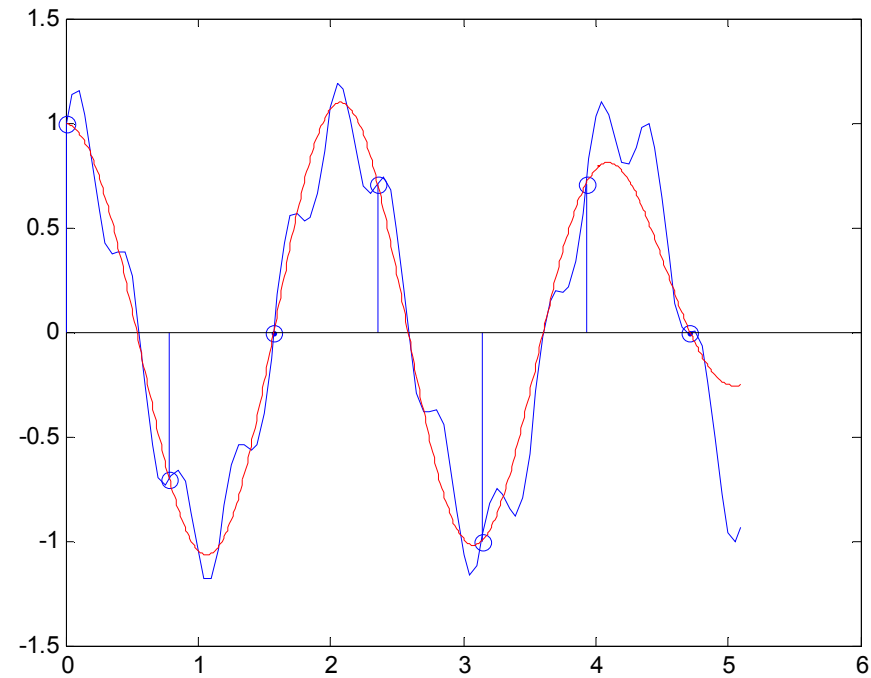
Παρεμβολή και το κριτήριο Nyquist

- Η παρεμβολή των τιμών ενός σήματος μπορεί να γίνει ιδανικά (με σφάλμα ίσο με 0) όταν έχει ικανοποιηθεί το κριτήριο του Nyquist:
 - η συχνότητα δειγματοληψίας είναι μεγαλύτερη από τη διπλάσια συχνότητα που περιέχει το σήμα
- Ακολουθεί παράδειγμα ανακατασκευής αναλογικού σήματος με:
 - f_s : συχνότητα δειγματοληψίας
 - f_a : μέγιστη συχνότητα αναλογικού σήματος,
 - T_s : περίοδος δειγματοληψίας
 - T_a ελάχιστη περίοδος αναλογικού σήματος

Παραδείγματα DAC



DAC $f_s = f_a/2$

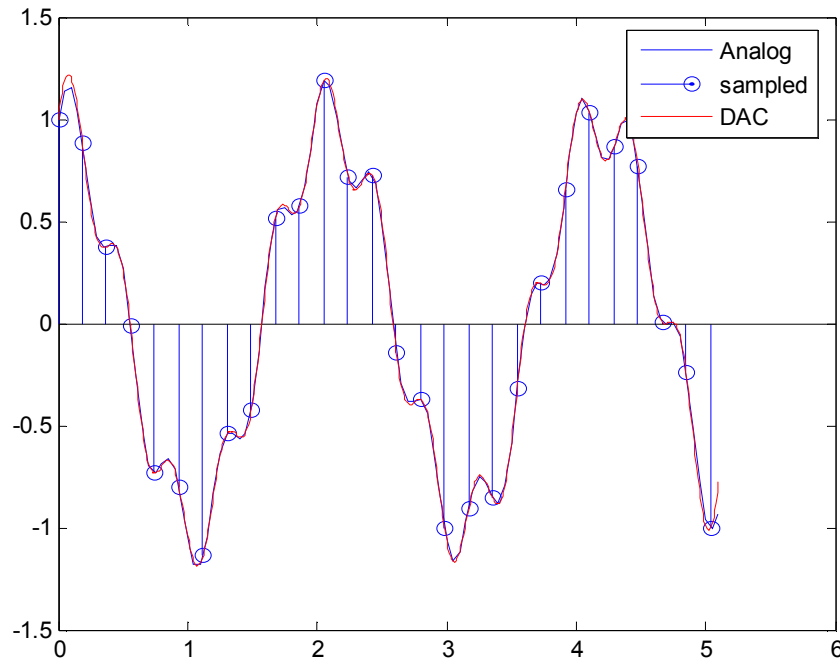


DAC $T_s = 19 \cdot T_a/40$

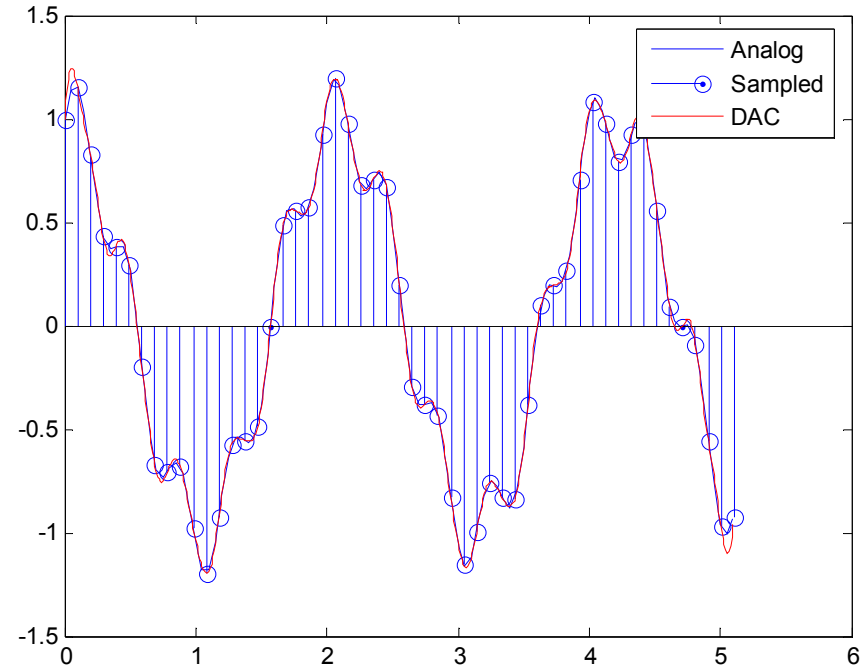
Δε πληρείται το κριτήριο Nyquist

Η κόκκινη καμπύλη είναι λανθασμένη ανακατασκευή

Παραδείγματα DAC



$$\text{DAC } f_s = f_a/4, T_s = 4 * T_a$$



$$\text{DAC } f_s = 4 * f_a, T_s = T_a/4$$

Ικανοποιείται το κριτήριο Nyquist

Η κόκκινη καμπύλη αποτελεί ορθή ανακατασκευή

Η παρεμβολή σαν συνέλιξη με πυρήνα

- Εστω διακριτό σήμα $x(n)$ και πυρήνας $h(n)$. Η τιμή στη θέση t δίνεται:

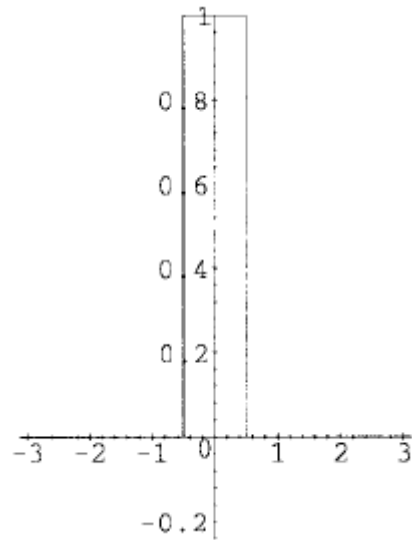
$$x(t) = \sum_{k=0}^N x(k)h(t-k)$$

- Υπάρχει πλήθος πυρήνων $h(t)$ που δρουν σαν παρεμβολείς
- Βασικό πλεονέκτημα το μικρό πλήθος σημείων ($N+1$)

Πυρήνες για απλά είδη παρεμβολής

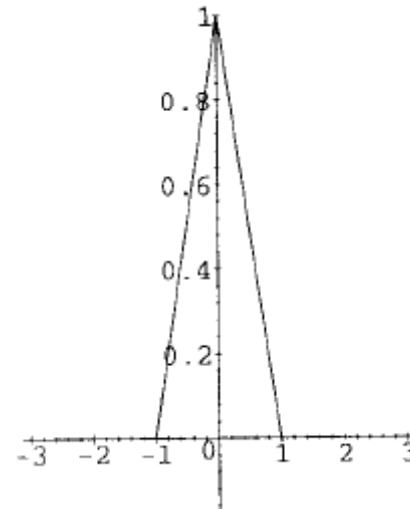
Παρεμβολή κοντινότερου γείτονα (nearest neib. interpolation)

$$h_1(x) = \begin{cases} 1, & 0 \leq |x| < 0.5 \\ 0, & \text{elsewhere.} \end{cases}$$

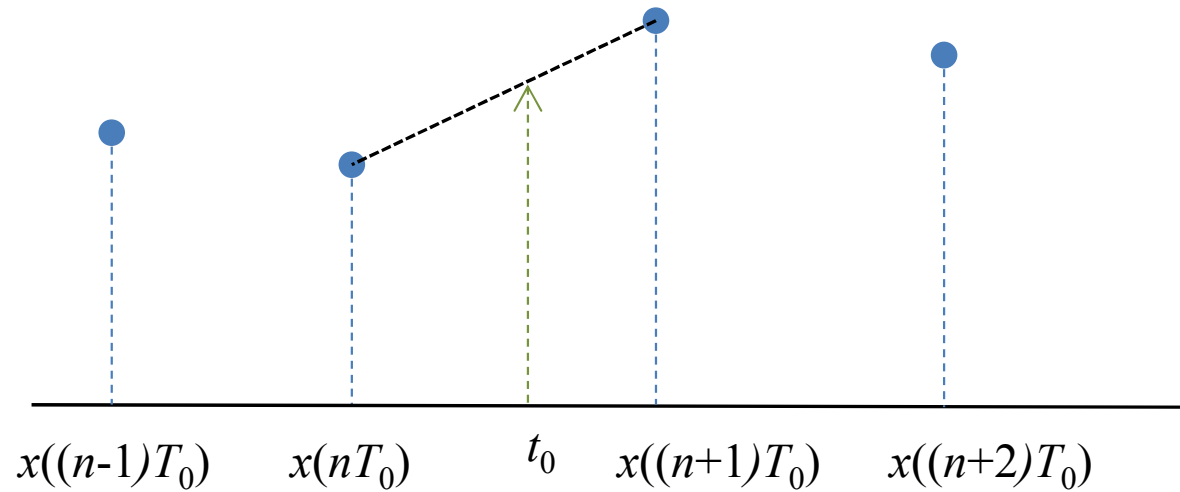


Γραμμική παρεμβολή (linear interpolation)

$$h_2(x) = \begin{cases} 1 - |x|, & 0 \leq |x| < 1 \\ 0, & \text{elsewhere.} \end{cases}$$



$$y_0 = y_n \frac{x_{n+1} - t_0}{x_{n+1} - x_n} + y_{n+1} \frac{t_0 - x_n}{x_{n+1} - x_n}$$



Γραμμική παρεμβολή του σήματος στο $t=t_0$

Πολυωνυμική παρεμβολή

- Πολύ συνηθισμένη παρεμβολή (πχ Lagrange)
- Συνήθως γίνεται κατά τμήματα (ομάδες δειγμάτων, διότι πολυώνυμο που παρεμβάλλει N σημεία έχει βαθμό $N-1$. Έτσι για $N > 5$ το πολυώνυμο γίνεται ασταθές
- Πιο συχνή είναι η κυβική παρεμβολή

Κυβική Παρεμβολή

- Για κάθε σημείο t_0 στο οποίο απαιτείται παρεμβολή:
 - Βρίσκονται 2 προηγούμενα και τα 2 επόμενα σημεία του σήματος, εστω n_0-1, n_0 και n_0+1, n_0+2
 - Πχ. Αν ζητάμε παρεμβολή για $t=10.2$, τότε χρησιμοποιούμε τα δείγματα 9, 10, 11 και 12.
 - Εξισώνοντας τις τιμές του σήματος στις 4 θέσεις, με τις τιμές ενός πολυωνύμου 3^{ου} βαθμού, υπολογίζουμε τους 4 συντελεστές του πολυωνύμου
 - Χρησιμοποιούμε το πολώνυμο για να υπολογίσουμε την τιμή του σήματος στη ζητούμενη θέση

Κυβική παρεμβολή με χρήση πυρήνα (kernel)

- Αποδεικνύεται ότι τα προηγούμενα ισοδυναμούν με τη συνέλιξη του σήματος με τον ακόλουθο πυρήνα (η παράμετρος a καθορίζει μία οικογένεια πυρήνων με παρόμοιες ιδιότητες)

$$K_{cubic}(t) = \begin{cases} (a+2)|t|^3 - (a+3)|t|^2 + 1, & |t| < 1 \\ a|t|^3 - 5a|t|^2 + 8a|t| - 4a, & 1 \leq |t| < 2 \end{cases}$$

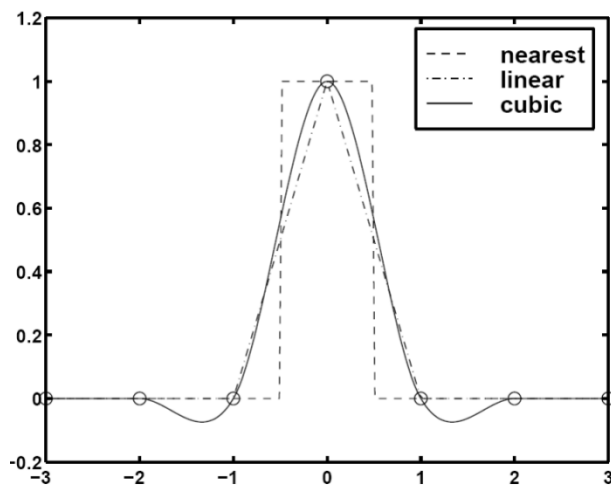
- Συχνά χρησιμοποιείται ο πυρήνας για $a=-1/2$

$$K_{cubic}(t) = \begin{cases} \frac{3}{2}|t|^3 - \frac{5}{2}|t|^2 + 1, & |t| < 1 \\ -\frac{1}{2}|t|^3 + \frac{5}{2}|t|^2 - 4|t| + 2, & 1 \leq |t| < 2 \end{cases}$$

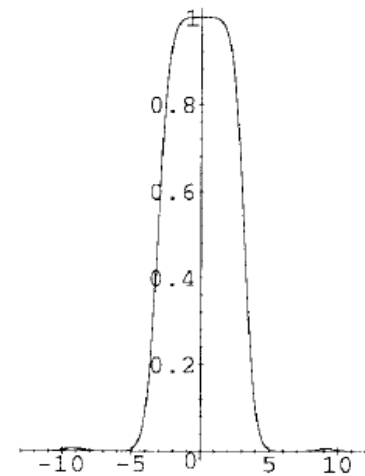
- Για οποιοδήποτε t_0 (με μη ακέραια τιμή), n_0 ο μικρότερος ακέραιος πλησιέστερος στο n_0 .
- Υπολογίζεται η τιμή του πυρήνα στις θέσεις $(t_0 - (n_0 - 1))$, $(t_0 - n_0)$, $(t_0 - (n_0 + 1))$, $(t_0 - (n_0 + 1))$
- Υπολογίζεται η ζητούμενη τιμή από τη σχέση

$$x(t) = \sum_{k=0}^3 x(k) K(t-k)$$

- Ο πυρήνας κυβικής παρεμβολής και η απόκριση συχνότητας (προκύπτει από το μετασχηματισμό Fourier)
- Παρατηρούμε ότι ο πυρήνας αποτελεί Lowpass φίλτρο
- Το σημαντικό πλεονέκτημα είναι ότι ο πυρήνας έχει μόνο 4 σημεία στήριξης (έναντι των απείρων του ιδανικού πυρήνα sinc) και απόκριση συχνότητας παρόμοια με το ιδανικό



DFT



Παρεμβολή πολυδιάστατων σημάτων

- Οι προηγούμενοι 1D πυρήνες εφαρμόζονται σε περισσότερες διαστάσεις, διαδοχικά (πχ σε 2D πρώτα κατά γραμμές και επί του αποτελέσματος εφαρμόζονται κατά στήλες)
- Ισοδύναμα Ο 2D πυρήνας $h(x,y)$ είναι διαχωρίσιμος

$$h_2(x, y) = h(x)h(y)$$

- Ο 2D πυρήνας $h(x,y)$ είναι διαχωρίσιμος:

$$I(x, y) = \sum_{k=0}^N I(k, l) h_2(x - k, y - l)$$

Παρεμβολή εικόνας

- Εστω ότι δεδομένης μίας εικόνας I_1 $N \times M$ θέλουμε να δημιουργήσουμε εικόνα I_2 διαστάσεων $2N \times 2M$. Οι τιμές των νέων pixel της I_2 υπολογίζονται βάσει παρεμβολής των τιμών των pixel της I_1 .
- Δύο είναι οι απλούστεροι τρόποι υπολογισμού των τιμών pixel της I_2 :
 - Κοντινότερο γείτονα: αναδιπλασιασμό (replication) τιμών (zero order hold κράτηση μηδενικής τάξης)
 - γραμμική παρεμβολή –linear interpolation- τιμών (first order hold κράτηση πρώτης τάξης)

Μεταβολή αριθμού pixel της εικόνας παρεμβολή κοντινότερου γείτονα

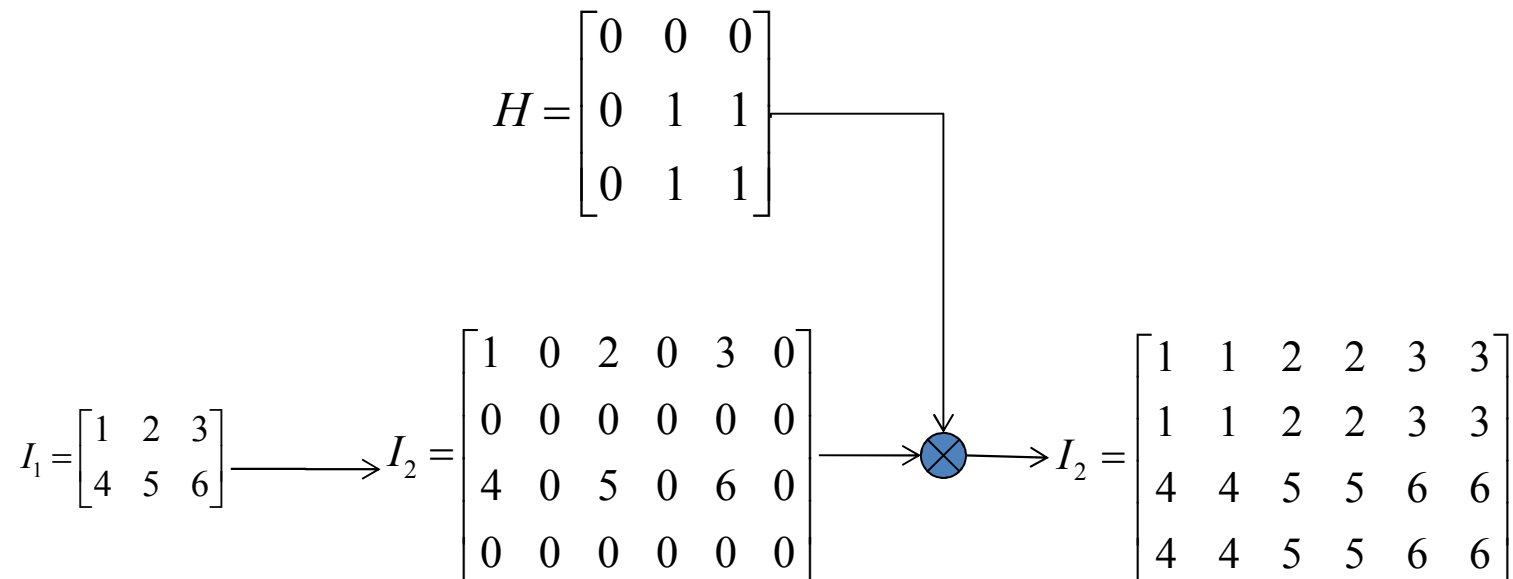
- Έστω διπλασιασμός της διάστασης της εικόνας: $I_1: M \times N \rightarrow I_2: 2M \times 2N$:
- Διπλασιάζουμε τις διαστάσεις της εικόνας συμπληρώνοντας τα νέα pixel με μηδενικές τιμές, δηλ. θέτουμε:

$$I_2(i, j) = \begin{cases} I_1\left(\frac{i}{2}, \frac{j}{2}\right), & i \text{ και } j : \text{άρτια} \\ 0 & i \text{ ή } j : \text{περιττό} \end{cases}$$

- Για κάθε ένα από τα 3 μηδενικά γειτονικά pixel του (i, j) με i, j άρτια, θέτουμε τιμή ίση με $I_1(i, j)$, σύμφωνα με τον παρακάτω ψευδοκώδικα:
- ```
for i=1:M-1
 - for j=1:N-1
 • I2(2i, 2j) = I1(i, j)
 • I2(2i+1, 2j) = I1(i, j)
 • I2(2i, 2j+1) = I1(i, j)
 • I2(2i+1, 2j+1) = I1(i, j)
 - end
• end
```

Ο συγκεκριμένος αλγόριθμος γενικεύεται για νέα διάσταση τυχαίο ακέραιο πολλαπλάσιο των  $M \times N$ .

- Τα προηγούμενα βήματα ισοδυναμούν με συνέλιξη της  $I_2$  με την μάσκα  $H$
- Παράδειγμα:



## Η γραμμική παρεμβολή σε εικόνες

- Εστω ότι επιθυμούμε να υπολογίσουμε την τιμή μίας εικόνας σε ένα σημείο με μη ακέραιες συντεταγμένες  $(y,x) = ([y]+a, [x]+b)$ , όπου  $[x], [y]$  το ακέραιο μέρος των  $x, y$  και  $a, b$  το κλασματικό τους μέρος.

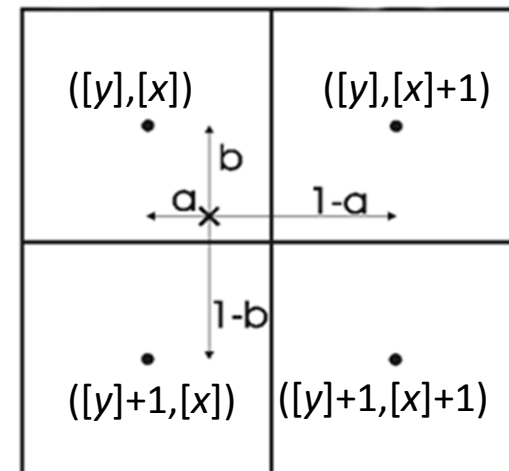
$$x = [x] + b$$

$$y = [y] + a$$

$$[x], [y] \in \mathbb{N}, a, b \in \mathbb{Q}^+, 0 < a, b < 1$$

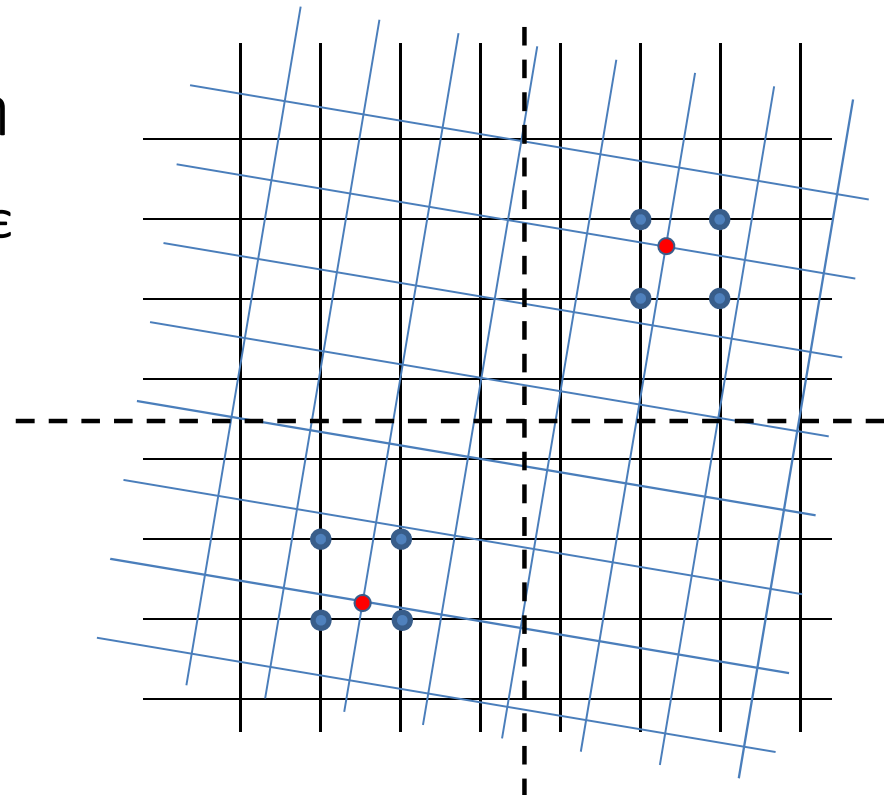
- Η τιμή της  $I$  στη θέση  $(y,x)$  είναι ο μέσος όρος των 4 γειτονικών pixel που το περικλείουν, σταθμισμένος με τον αντίστροφο της απόστασής του  $(y,x)$  από τα κέντρα τους:

$$I([y]+a, [x]+b) = (1-a)(1-b)I([y], [x]) + a(1-b)I([y], [x]+1) \\ + (1-a)bI([y]+1, [x]) + abI([y]+1, [x]+1)$$



# Περιστροφή της εικόνας και η εφαρμογή της παρεμβολής

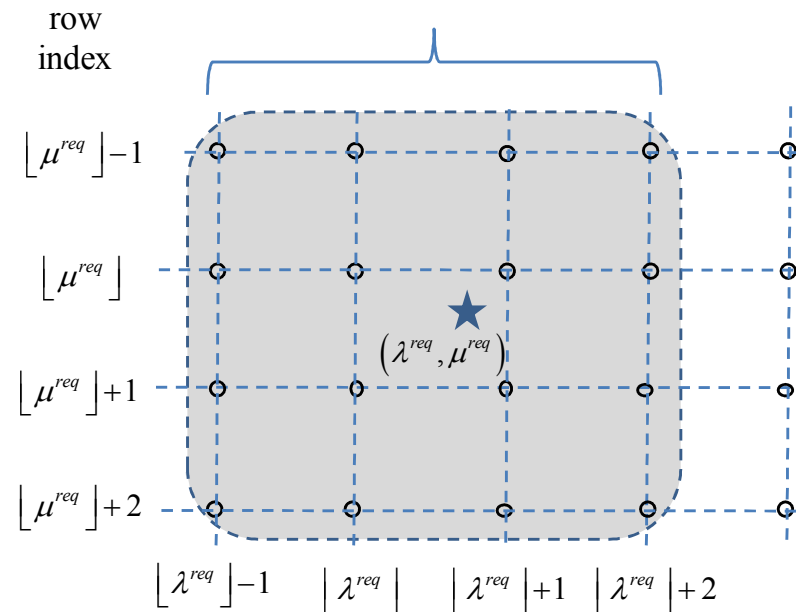
- Εστω αρχική εικόνα  $I(x,y)$  (μαύρο πλέγμα γραμμών) και η περιστραμμένη γύρω από το κέντρο της  $I_1(x,y)$  (πλέγμα μπλε γραμμών)
- Τα pixel είναι τα σημεία τομής των γραμμών του πλέγματος
- Τα pixel της  $I_1$  παίρνουν τιμές από τα κοντινότερα Pixel της  $I$ .
- Η εύρεση των τιμών της  $I_1$  γίνεται με παρεμβολή της  $I$



# Δικυβική παρεμβολή σε εικόνες

- Η εφαρμογή της κυβικής παρεμβολής κατά γραμμές και στη συνέχεια κατά στήλες ονομάζεται δικυβική
- Ο πυρήνας της δικυβικής παρεμβολής είναι 4x4

Για παρεμβολή της εικόνας σε 1 σημείο απαιτούνται 4x4 σημεία στήριξης 2 πριν και 2 μετά το ζητούμενο σημείο, κατά γραμμές και κατά στήλες





## Μεταβολή αριθμού pixel της εικόνας με γραμμική παρεμβολή

- Βήμα 1: Δημιουργείται μία νέα εικόνα με το ζητούμενο μέγεθος, σύμφωνα με τα προηγούμενα. Τα επιπλέον pixel έχουν τιμή 0.

$$I_2(i, j) = \begin{cases} I_1\left(\frac{i}{2}, \frac{j}{2}\right), & \text{i και j : άρτια} \\ 0 & \text{i ή j : περιττό} \end{cases}$$

- Βήμα 2: Γραμμική παρεμβολή κατά γραμμές:

$$I_2(2i+1, 2j) = \frac{1}{2}(I_1(i+1, j) + I_1(i, j))$$

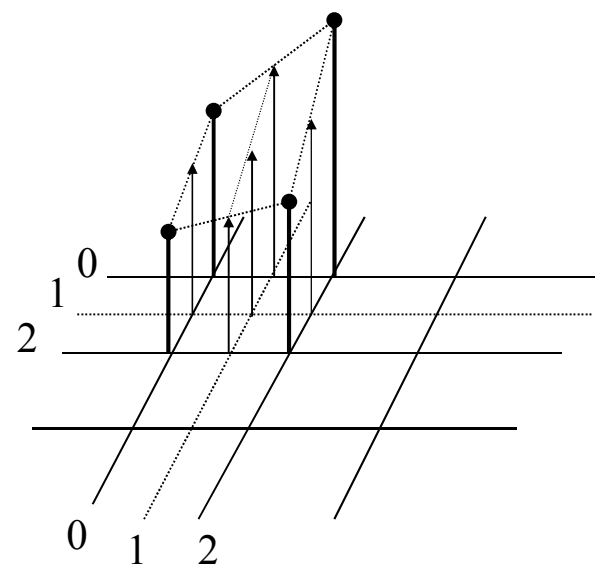
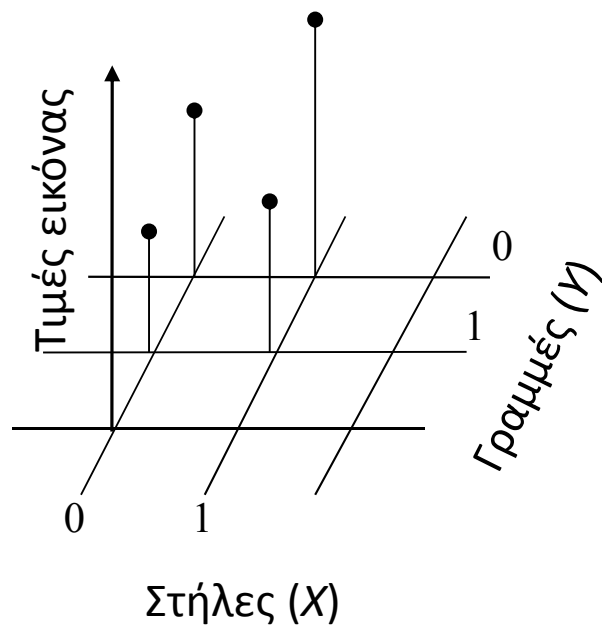
- Βήμα 3: Γραμμική παρεμβολή κατά στήλες:



- Προσοχή: η παρεμβολή κατά στήλες γίνεται και για τις νέες στήλες που δημιουργήθηκαν από το βήμα 1.

$$I_2(2i, 2j+1) = \frac{1}{2}(I_1(i, j+1) + I_1(i, j))$$

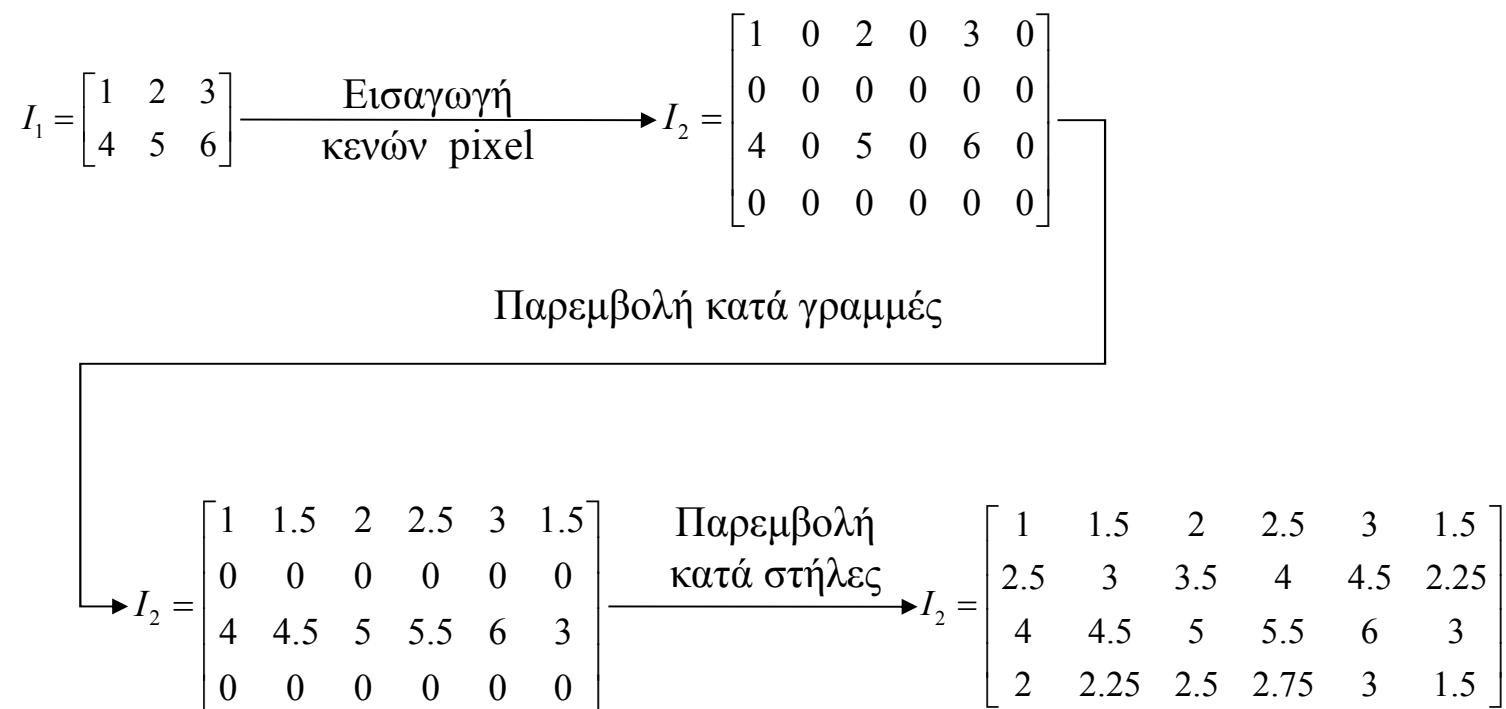
Ο συγκεκριμένος αλγόριθμος γενικεύεται για νέα διάσταση και για τυχαίο ακέραιο πολλαπλάσιο των  $M \times N$ .

# Η έννοια της γραμμικής παρεμβολής εικόνας 2D



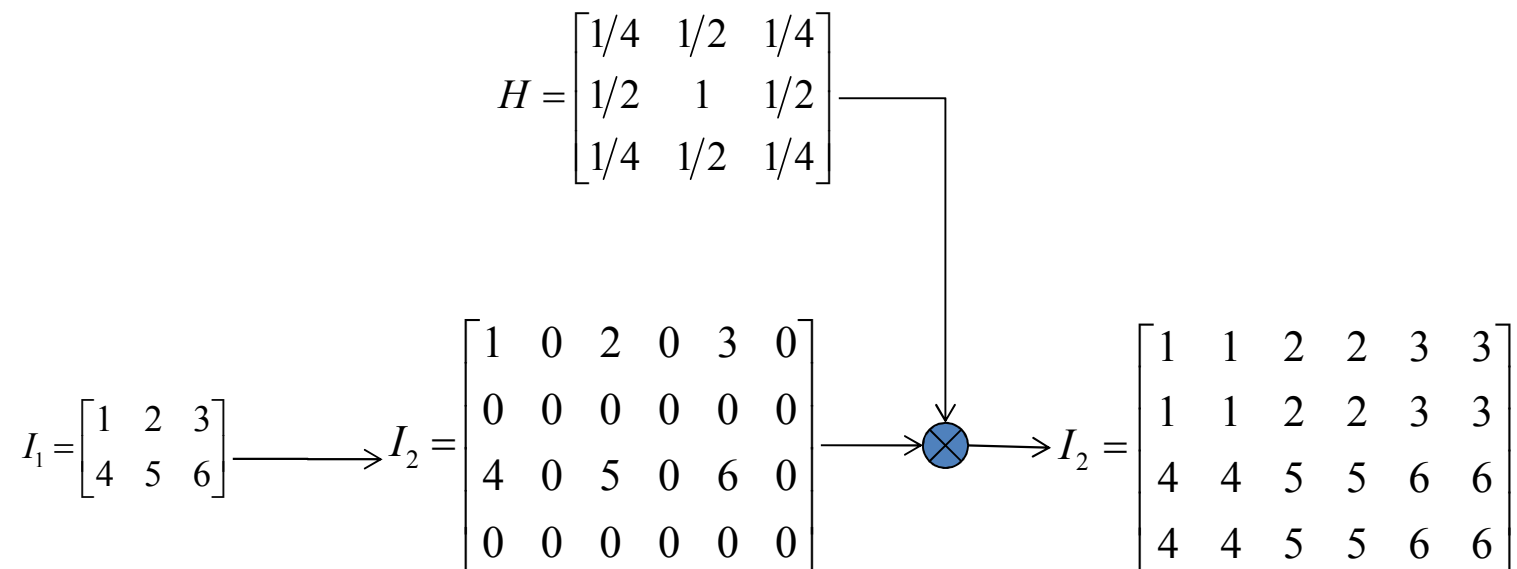
-  Αρχική τιμή εικόνας
-  Τιμή εικόνας από παρεμβολή

## Παράδειγμα γραμμικής παρεμβολής εικόνας



Με διαδοχική εφαρμογή πρώτα κατά γραμμές και μετά κατά στήλες

- Τα προηγούμενα βήματα ισοδυναμούν με συνέλιξη της  $I_2$  με την μάσκα  $H$



# Σύγχρονες Μέθοδοι Παρεμβολής: State of the Art και Βιβλιογραφία

- Γενικευμένη παρεμβολή (generalized interpolation)
  - Οικογένεια b-spline, O-MOMs κλπ: οι πυρήνες αυτοί δε συνελίσσονται απευθείας με τις τιμές του σήματος, αλλά με συντελεστές που υπολογίζονται κατάλληλα

- Παρεμβολή με χρήση παραγώγων:
  - παράγονται συναρτήσεις, οι οποίες όχι μόνο παρεμβάλουν τις τιμές του σήματος, αλλά και τις τιμές των παραγώγων του σήματος, οποιασδήποτε τάξης
  - Οι παράγωγοι υπολογίζονται αριθμητικά με κεντρικές διαφορές, ή implicitly

- Μέθοδοι παρεμβολής ιατρικών εικόνων
  - Lehmann, T. M., Gönner, C., & Spitzer, K. (1999). Survey: Interpolation methods in medical image processing, *IEEE Transactions on Medical Imaging*, 18(11), 1049-1075
- Γενικευμένη παρεμβολή (b-spline κλπ)
  - Infinite impulse response (IIR) *prefiltering* algorithm described in the work of Unser et al [IEEE T. SP, 41, 1993]
  - T. Blu, P. Thévenaz, and . Unser, Complete Parameterization of Piecewise-Polynomial Interpolation Kernels, IEEE TRANSACTIONS on Image Processing, 12(11), 2003
  - [http://bigwww.epfl.ch/tutorials/unser\\_isbi\\_06\\_part1.pdf](http://bigwww.epfl.ch/tutorials/unser_isbi_06_part1.pdf)

- Παρεμβολή με χρήση παραγώγων (Hermite)
  - Delibasis, K. K., & Kechriniotis, A. (2014). A new formula for bivariate hermite interpolation on variable step grids and its application to image interpolation. *Image Processing, IEEE Transactions on*, 23(7), 2892-2904.
  - Delibasis, K. K., Kechriniotis, A. I., & Assimakis, N. D. (2012). New closed formula for the univariate hermite interpolating polynomial of total degree and its application in medical image slice interpolation. *Signal Processing, IEEE Transactions on*, 60(12), 6294-6304
  - Delibasis, K. K., Kechriniotis, A., & Maglogiannis, I. (2013). On centered and compact signal and image derivatives for feature extraction. In *Artificial Intelligence Applications and Innovations* (pp. 318-327). Springer Berlin Heidelberg.



## Απλός αλγόριθμος εφαρμογής συσχετισμένου μετασχηματισμού: μη ενδεικνυόμενος

Εστω ότι θέλουμε να μετασχηματίσουμε γεωμετρικά δοθείσα εικόνα  $I_1$ , βάσει συσχετισμένου μετασχηματισμού με πίνακα  $A$ .

Βήμα 1ο: Προσδιορίζουμε την προβολή (απεικόνιση) του κέντρου του κάθε pixel  $(x,y)$  της αρχικής εικόνας  $I_1$  στη νέα εικόνα  $I_2$ :  
 $(x_1, y_1, 1)^T = A * (x, y, 1)^T$ .

Βήμα 2ο: Βρίσκουμε το pixel της νέας εικόνας  $I_2$  του οποίου το κέντρο βρίσκεται πιο κοντά στο σημείο της προβολής (στρογγυλοποίηση συντεταγμένων)

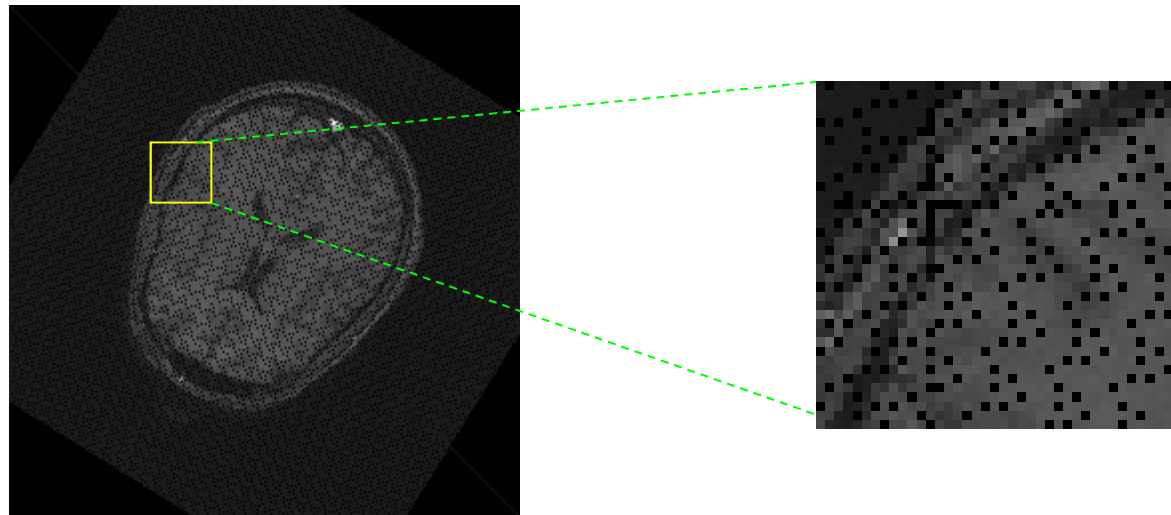
Βήμα 3ο: Θέτουμε τη φωτεινότητα του ζητούμενου pixel της τελικής εικόνας ίση με τη φωτεινότητα του pixel, από το βήμα 2, της αρχικής εικόνας:

$$I_2(\text{round}(x_1), \text{round}(y_1)) = I_1(x, y) .$$

Παράδειγμα ψευδοκώδικα του προηγούμενου αλγόριθμου για περιστροφή εικόνας κατά γωνία  $\theta$  γύρω από το κέντρο μάζας της CM.

```
A=T(-CM)R(θ)T(CM)
for i=1:256
 for j=1:256
 [i1,j1]T=A*[i,j]T
 if i1>256 →i1=256;
 if i1<=1→i1=1;
 if j1>256 →j1=256;
 if j1<=1→j1=1;
 IM(round(i1),round(j1))=I(i,j);
 end;
end;
```

## Παράδειγμα περιστροφής εικόνας με χρήση του προηγούμενου αλγόριθμου



Παρατηρούμε το artifact των μηδενικών Pixel (τα οποία δεν έχουν πάρει τιμές). Η χρήση της παρεμβολής βάσει του κοντινότερου γείτονα είναι πολύ απλή στην υλοποίηση αλλά έχει το μειονέκτημα ότι θολώνει την περιστραμμένη εικόνα. Εναλλακτικά μπορεί να χρησιμοποιηθεί η διγραμμική παρεμβολή.

# Γενικός αλγόριθμος εφαρμογής συσχετισμένου μετασχηματισμού

- Βήμα 1ο: Προσδιορίζουμε τη προβολή (απεικόνιση) του κέντρου του κάθε pixel  $(x,y)$  της νέας εικόνας  $I_2$  στην αρχική εικόνα  $I_1$ :  $(x_1, y_1, 1)^T = A^{-1} * (x, y, 1)^T$ .

Υπολογίζουμε την τιμή του  $(x,y)$  της νέας εικόνας  $I_2$  με 2 τρόπους:

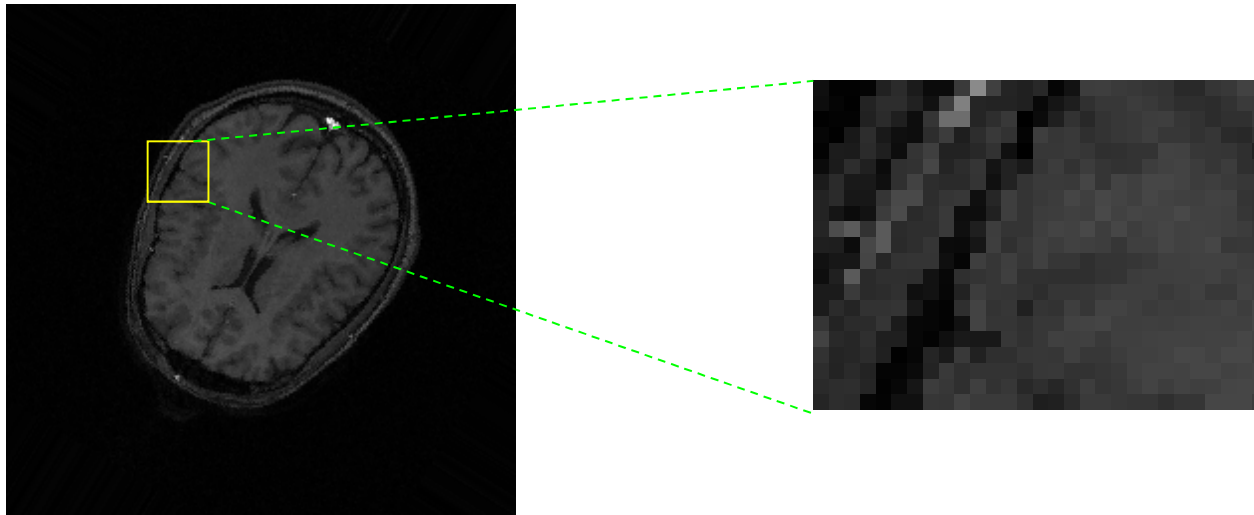
- Βήμα 2α: **Παρεμβολή κοντινότερου γείτονα**: Βρίσκουμε το pixel της αρχικής εικόνας του οποίου το κέντρο βρίσκεται πιο κοντά στο σημείο της προβολής με στρογγυλοποίηση:  $I_2(x,y) = I_1(\text{round}(x_1), \text{round}(y_1))$
- Βήμα 2β: **Διγραμμική Παρεμβολή**: Θέτουμε την τιμή του  $(x_1, y_1)$  ζητούμενου pixel της τελικής εικόνας  $I_2$ , ίση με την τιμή της αρχικής εικόνας  $I_1$  στη θέση  $(x,y)$ . Επειδή τα  $(x_1, y_1)$  είναι εν γένει πραγματικοί αριθμοί, εφαρμόζουμε διγραμμική παρεμβολή:

$$I_2(y, x) = I_1(y_1, x_1) = I_1([y_1] + a, [y_1] + b) = \\ (1-a)(1-b)I([y_1], [x_1]) + a(1-b)I([y_1], [x_1] + 1) + (1-a)bI_1([y_1] + 1, [x_1]) + abI_1([y_1] + 1, [x_1] + 1)$$

# Αλγόριθμος περιστροφής εικόνας για διόρθωση artifact, βάσει του κοντινότερου γείτονα

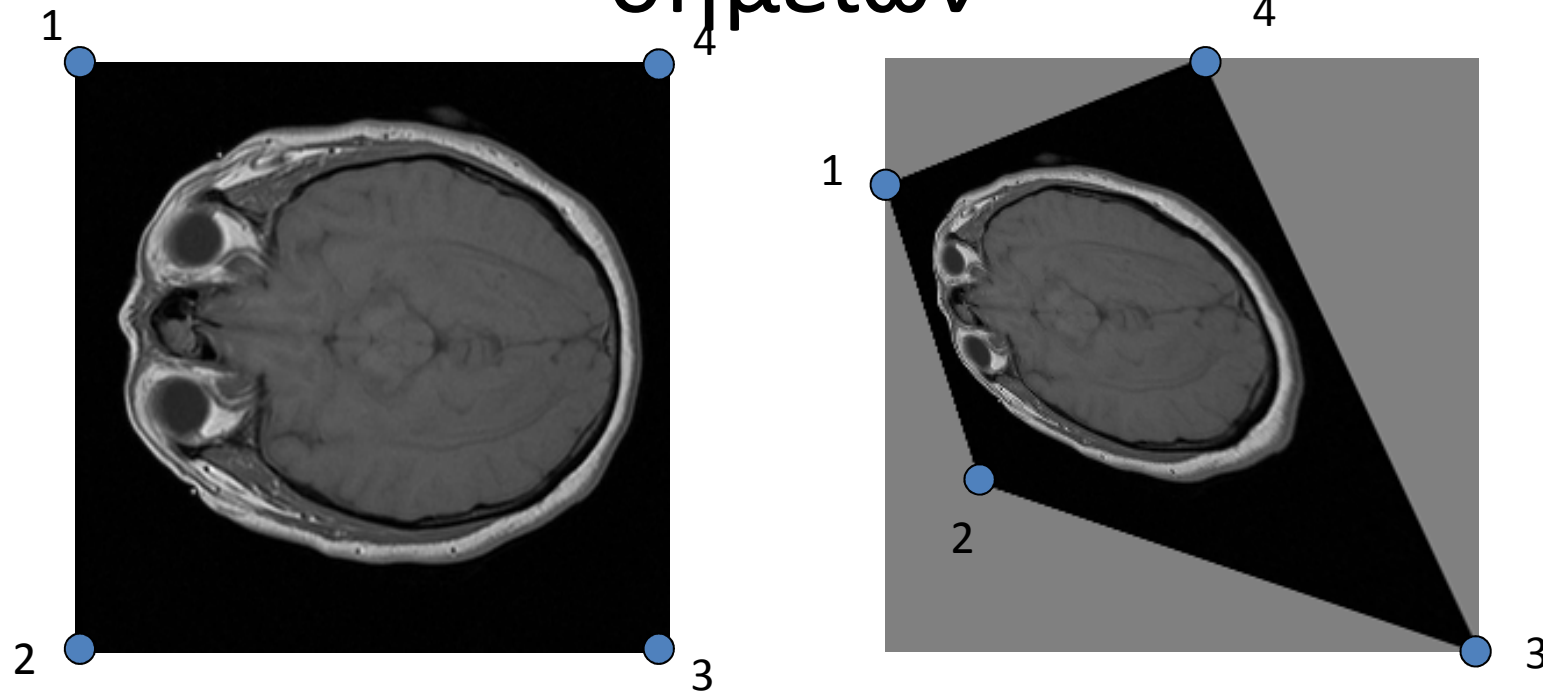
```
A=T (-CM) R (θ) T (CM) // Υπολογισμός του πίνακα περιστροφής A
for i=1:256 // Για κάθε pixel της νέας εικόνας I2
 for j=1:256
 [i1, j1]T=(A-1) * [i, j]T // Υπολογισμός του pixel της I1 από το οποίο
 // προέρχεται με χρήση του αντίστροφου του A
 if i1>256 →i1=256;
 if i1<=1→i1=1; // Ελεγχος αν το (i1,j1) είναι εντός
 if j1>256 →j1=256; // της αρχικής εικόνας
 if j1<=1→j1=1;
 IM2 (i, j)=
 IM(round(i1) , round(j1)) ; // Μέθοδος του κοντινότερου γείτονα
 end;
end;
```

# Παράδειγμα περιστροφής εικόνας χωρίς artifacts με χρήση του προηγούμενου αλγόριθμου



Παρατηρούμε ότι το artifact των μηδενικών Pixel έχει διορθωθεί. Η χρήση της παρεμβολής βάσει του κοντινότερου γείτονα είναι πολύ απλή στην υλοποίηση αλλά έχει το μειονέκτημα ότι θολώνει την περιστραμμένη εικόνα. Εναλλακτικά μπορεί να χρησιμοποιηθεί η διγραμμική παρεμβολή.

# Παραμόρφωση εικόνας με χρήση 4 σημείων



- κξσφκσδξφλ

- Γενική μορφή μετασχηματισμού
 
$$x_1 = F_1(x, y)$$

$$y_1 = F_2(x, y)$$
- Διγραμμικός (δεν διατηρεί ευθείες)
 
$$x_1 = a_0 + a_1x + a_2y + a_3xy$$

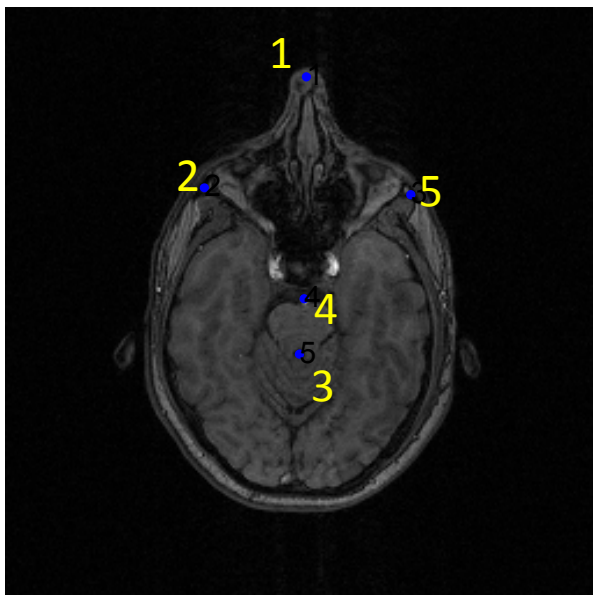
$$y_1 = b_0 + b_1x + b_2y + b_3xy$$
- Προβολικός (διατηρεί ευθείες)
 
$$x_1 = \frac{a_0 + a_1x + a_2y}{1 + c_1x + c_2y}, y_1 = \frac{b_0 + b_1x + b_2y}{1 + c_1x + c_2y}$$
- Οι παράμετροι του μετασχηματισμού υπολογίζονται από τα 4 σημεία με τις μετατοπίσεις τους



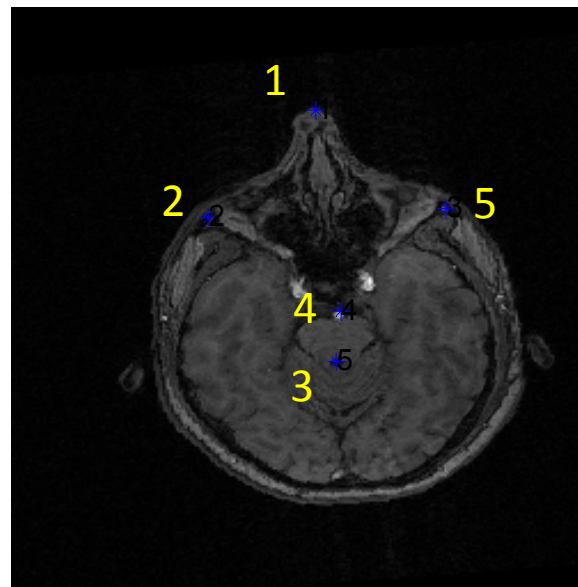
## Χωρική ταύτιση εικόνων: Καθορισμός του πίνακα μετασχηματισμού βάσει ομόλογων σημείων

- Το πρόβλημα της χωρικής ταύτισης θεωρεί ότι υπάρχουν 2 εικόνες  $I_1, I_2$  του ίδιου αντικειμένου, που έχουν συλλεχθεί υπό διαφορετική γεωμετρία. Εστω ότι υπάρχει ένας αριθμός από ζεύγη **ομόλογων** σημείων μεταξύ δύο εικόνων  $I_1, I_2$ :  $\{p_i^A\}$  στην  $I_1$  και  $\{p_i^B\}$  στην  $I_2$ . Ζητείται ο πίνακας του μετασχηματισμού που μετασχηματίζει γεωμετρικά την  $I_1$  στην  $I_2$ , έτσι ώστε τα μετασχηματισμένα σημεία  $\{p_i^A\}$  να συμπίπτουν με τα σημεία  $\{p_i^B\}$ . Λέμε τότε ότι οι δύο εικόνες ταυτίζονται χωρικά (spatial registration).
- Τα ζεύγη ομόλογων σημείων  $\{p_i^A\}$  στην  $I_1$  και  $\{p_i^B\}$  στην  $I_2$ ,  $i=1, \dots, N$   $N>3$ , ορίζονται είτε από το χρήστη είτε από κάποια αυτόματη μέθοδο.
- **Ομόλογα** είναι δύο σημεία όταν έχουν τοποθετηθεί πάνω στα ίδια αντικείμενα στις δύο διαφορετικές εικόνες. Προφανώς, επειδή οι δύο εικόνες δεν ταυτίζονται χωρικά, οι συντεταγμένες δύο ομόλογων σημείων δεν θα είναι ίδιες (πχ η μύτη του ασθενή στην  $I_1$  δεν βρίσκεται στα pixel στα οποία βρίσκεται η μύτη του ίδιου ασθενή στην  $I_2$ ).

Image I1



Transformed I1



- Παράδειγμα δύο εικόνων I1 και I2 του ίδιου αντικειμένου (MRI εγκεφάλου) με 5 ζεύγη ομολόγων σημείων που έχουν τοποθετηθεί σε κοινές ανατομικές δομές από τον χρήστη.

- Για να καθορίσουμε το μετασχηματισμό Affine χρειαζόμαστε τον πίνακα του μετασχηματισμού ο οποίος έχει 6 αγνώστους:

$$a_{00}, a_{01}, a_{02}, a_{10}, a_{11}, a_{12}$$

- Οι 3 άγνωστοι  $a_{00}, a_{01}, a_{02}$  υπολογίζονται από τις X συντεταγμένες των ζευγών ομολόγων σημείων και οι 3 άγνωστοι  $a_{10}, a_{11}, a_{12}$  υπολογίζονται από τις Y συντεταγμένες. Αν το πλήθος των ζευγών σημείων  $N=3$  πρέπει να επιλυθούν 2 γραμμικά συστήματα  $3 \times 3$ . Αν  $N > 3$  (συνήθης περίπτωση) τότε τα 2 γραμμικά συστήματα είναι υπερκαθορισμένα (για το κάθε ένα υπάρχουν 3 άγνωστοι και  $N > 3$  γραμμικές εξισώσεις).

- Κατασκευάστε τον πίνακα  $A = \begin{bmatrix} x_1^A & y_1^A & 1 \\ x_2^A & y_2^A & 1 \\ \dots & \dots & \dots \\ x_N^A & y_2^A & 1 \end{bmatrix}$

- Εστω  $p = (a_{00}, a_{01}, a_{02})^T$ ,  $q = (a_{10}, a_{11}, a_{12})^T$   
 $\mathbf{b}_1 = (x_1^A, x_2^A, \dots, x_N^A)^T$ ,  $\mathbf{b}_2 = (x_1^B, x_2^B, \dots, x_N^B)^T$

- Πρέπει να επιλυθούν τα γραμμικά συστήματα  $\mathbf{A}p = \mathbf{b}_1$   
 $\mathbf{A}q = \mathbf{b}_2$

- Θυμηθείτε ότι ο  $A$  είναι διαστάσεων  $N \times 3$  ενώ τα  $\mathbf{p}, \mathbf{q}$  είναι διαστάσεων  $3 \times 1$  και τα  $\mathbf{b}_1, \mathbf{b}_2$  είναι διαστάσεων  $N \times 1$ . Ο ευκολότερος τρόπος είναι να χρησιμοποιηθεί ο τελεστής «\» του Matlab:  $p = A \setminus \mathbf{b}_1$  και  $q = A \setminus \mathbf{b}_2$ .

- Η παραπάνω λύση ισοδυναμεί με τη μέθοδο ελαχίστων τετραγώνων, σύμφωνα με την οποία τα δύο συστήματα επιλύονται ως εξής:

$$(\mathbf{A}^T \mathbf{A}) \mathbf{q} = \mathbf{A}^T \mathbf{b}_2$$

# Ελαστικοί γεωμετρικοί μετασχηματισμοί

- Ο πιο γνωστός ελαστικός γεωμετρικός μετασχηματισμός είναι το μοντέλο TPS (Thin Plate Splines) (Bookstein 1989).

# Ελαστικοί γεωμετρικοί μετασχηματισμοί: το μοντέλο TPS (Thin Plate Splines)

- Εστω ότι έχουμε επιλέξει 2 σύνολα ομόλογων σημείων  $\{x_i, y_i\}$ ,  $\{x'_i, y'_i\}$
- Κατασκευάζουμε τους πίνακες  $P$ ,  $V$

$$P = \begin{bmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ \dots & \dots & \dots \\ 1 & x_n & y_n \end{bmatrix}, 3 \times n; \quad V = \begin{bmatrix} x'_1 & x'_2 & \dots & x'_n \\ y'_1 & y'_2 & \dots & y'_n \end{bmatrix}$$

- Ορίζουμε  $r_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$  και κατασκευάζουμε τον πίνακα  $K$ , όπου  $r_{ij}$  η απόσταση των σημείων  $i, j$ .

$$K = \begin{bmatrix} 0 & U(r_{12}) & \dots & U(r_{1n}) \\ U(r_{21}) & 0 & \dots & U(r_{2n}) \\ \dots & \dots & \dots & \dots \\ U(r_{n1}) & U(r_{n2}) & \dots & 0 \end{bmatrix}, n \times n;$$

- Κατασκευάζουμε τον πίνακα  $L$

$$L = \left[ \begin{array}{c|c} K & P \\ \hline P^T & O \end{array} \right], (n + 3) \times (n + 3)$$

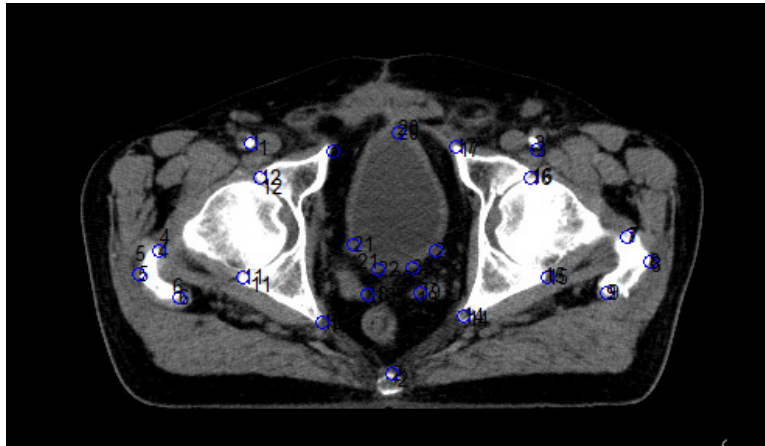
- Κατασκευάζουμε τον πίνακα  $Y = (V | 0 \ 0 \ 0)^T$

$$L^{-1}Y = (W | a_1 \ a_x \ a_y)^T$$

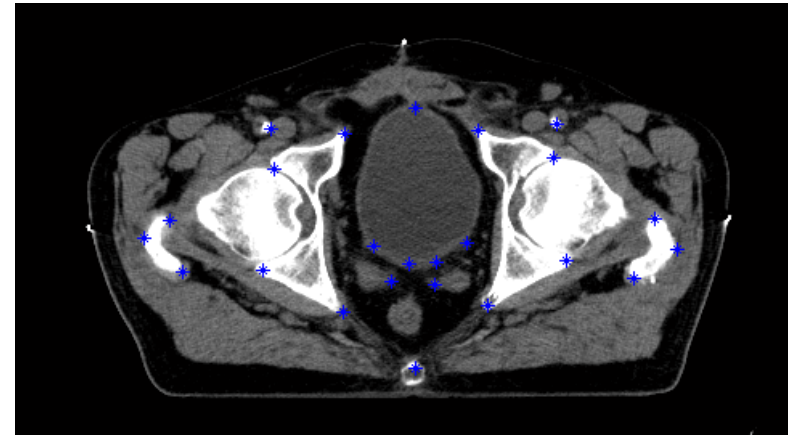
- Υπολογίζουμε για κάθε σημείο  $(x, y)$  τις νέες του συντεταγμένες βάση της ακόλουθης διανυσματικής συνάρτησης:

$$f(x, y) = a_1 + a_x x + a_y y + \sum_{i=1}^n w_i U(|P_i - (x, y)|)$$

# Παράδειγμα: ταύτιση εικόνων με ελαστικό μετασχηματισμό



(α)



(β)

- Εστω δύο εικόνες του ίδιου αντικειμένου που έχει παραμορφωθεί ελαστικά.
- Ορίζουμε ζεύγη ομόλογων σημείων στις εικόνες
- Αναζητούμε το μετασχηματισμό που παραμορφώνει την (α) ώστε να ταυτιστεί με την (β)



