

Computer vision

Σπύρος Σειμένης 5070

Δεκέμβριος 2014

1 Βασικοί Γεωμετρικοί Μετασχηματισμοί στη Matlab

1.1

- Η `imread` χρησιμοποιείται για να φορτωθεί ένα αρχείο εικόνας στη matlab.
- Η `imwarp` μετασχηματίζει μια δοσμένη εικόνα σύμφωνα με έναν μετασχηματισμό, και επιστρέφει εκτός από την νέα εικόνα και τις σχετικές συντεταγμένες της νέας εικόνας στον χώρο.
- Η `affine2d` δέχεται έναν διδιάστατο γεωμετρικό μετασχηματισμό `affine` και παράγει ένα αντικείμενο `affine2d` για χρήση με την `imwarp`.
- Η `projective2d` δέχεται έναν διδιάστατο γεωμετρικό μετασχηματισμό `projective` και παράγει ένα αντικείμενο `projective2d` για χρήση με την `imwarp`.
- Ένα αντικείμενο `imref2d` (επιστρέφεται από την `imwarp`) για μια εικόνα περιέχει την πληροφορία που χρειαζόμαστε για να την τοποθετήσουμε κατάλληλα μέσα στον κόσμο.
- Η `implay` δέχεται έναν πίνακα από εικόνες και τις αναπαράγει κατά σειρά

1.2



1.3

Το αποτέλεσμα φαίνεται στα αρχεία `shared_pudding.mat` και `pudding.m`

1.4

Το αποτέλεσμα φαίνεται στα αρχεία `transf_windmill.mat` και `windmill.m`.

Η προεπιλεγμένη μέθοδος `interpolation` είναι η `linear`. Όταν χρησιμοποιείται η μέθοδος `nearest` το αποτέλεσμα από τους μετασχηματισμούς έχει αρκετό θόρυβο κυρίως στις ακμές. Αυτό συμβαίνει διότι με την μέθοδο `nearest` κατά τον μετασχηματισμό τα `pixels` παίρνουν την τιμή αποκλειστικά του κοντινότερου γείτονα με αποτέλεσμα να δημιουργείται αυτό το σκαλωτό μοτίβο στις ακμές όπου το φόντο διαφέρει.

Το `cubic interpolation` δεν παρουσιάζει σκαλωτό μοτίβο στις ακμές όπως η `linear`. Αν και στην `linear` δεν είναι τόσο εμφανές όσο στην `nearest` εξακολουθεί να υπάρχει αυτή η παραμόρφωση. Η `cubic` επιτυγχάνει το πιο ομαλό αποτέλεσμα αλλά ο χρόνος που χρειάζεται το πρόγραμμα για να εκτελεστεί είναι μεγαλύτερος.

1.5

Όπως και στις προηγούμενες περιπτώσεις για την κίνηση της μπάλας χρησιμοποιήθηκε σταθερός αριθμός από `frames`. Σε κάθε `frame` για να μετακινείται η μπάλα γίνονται οι εξής μετασχηματισμοί με αυτή την σειρά:

1. Κλιμάκωση της μπάλας στο $1/4$ του μεγέθους της.
2. Αντίθετα με την περίπτωση του μύλου όπου γίνεται επιτόπου περιστροφή και μετακίνηση κατάλληλα, για να γίνουν πιο εύκολα οι πράξεις σε αυτή την περίπτωση μετακινώ την μπάλα στην αρχή των αξόνων.
3. Εκτελείται περιστροφή στην μπάλα με σταθερή γωνιακή ταχύτητα
4. Στην συνέχεια για σταθερά διαστήματα για τον άξονα των x υπολογίζεται η τιμή μιας συνάρτησης συνημιτόνου η οποία μεταβάλλεται ώστε να έχει πλάτος όσο το ύψος της εικόνας, αντιστρέφεται και μετατοπίζεται ώστε να λειτουργεί για τις αρνητικές συντεταγμένες για το y άξονα. Στις τιμές της μετακίνησης προστίθενται οι τιμές που επιστρέφει το αντικείμενο `imref2d` που επιστράφηκε από την προηγούμενη περιστροφή. Αυτό γίνεται ώστε η μετακίνηση της μπάλας να μην επηρεάζεται από τις περιστροφές της.
5. Έπειτα εκτελείται `padding` στην υπολογιζόμενη εικόνα ώστε να φτάσει σε μέγεθος το φόντο. Επίσης εκτελείται και `cropping` ώστε η εξομοίωση να δουλεύει ακόμα και όταν το μέγεθος της μπάλας είναι πολύ μεγάλο.

Για να μετακινείται η μπάλα προς τον ορίζοντα έγιναν οι εξής αλλαγές στον παραπάνω αλγόριθμο:

1. Η κλιμάκωση της μπάλας μειώνεται με γραμμικό τρόπο ώστε να δίνει την αίσθηση του βάθους καθώς μετακινείται. Εδώ θα μπορούσε να επιτευχθεί καλύτερη προσέγγιση χρησιμοποιώντας διαφορετικό τρόπο για την κλιμάκωση της. Δεν υπολογίστηκε με ακρίβεια το τρισδιάστατο δάπεδο που δημιουργεί η θάλασσα του φόντου ώστε να βρεθεί η ακριβής συνάρτηση κλιμάκωσης.
2. Μικρές αλλαγές στην συχνότητα του συνημιτόνου και της απόσβεσης ώστε να είναι πιο αισθητή η κίνηση με βάθος.
3. Η σημαντικότερη αλλαγή είναι στις γραμμές 62-67 όπου χρησιμοποιείται η `projective2d` για να μετασχηματίσει την συνάρτηση συνημιτόνου που χρησιμοποιήθηκε ώστε να μετακινεί την μπάλα προς το βάθος και δεξιά αντί για μόνο δεξιά. Για τον ίδιο λόγο διότι δεν υπολογίστηκε με ακρίβεια η γεωμετρία της θάλασσας η συνάρτηση δεν μετασχηματίστηκε με τον καλύτερο δυνατό τρόπο. Όλες οι αλλαγές φαίνονται στα αρχεία `transf_beach2.mat` και `ball_far.m`