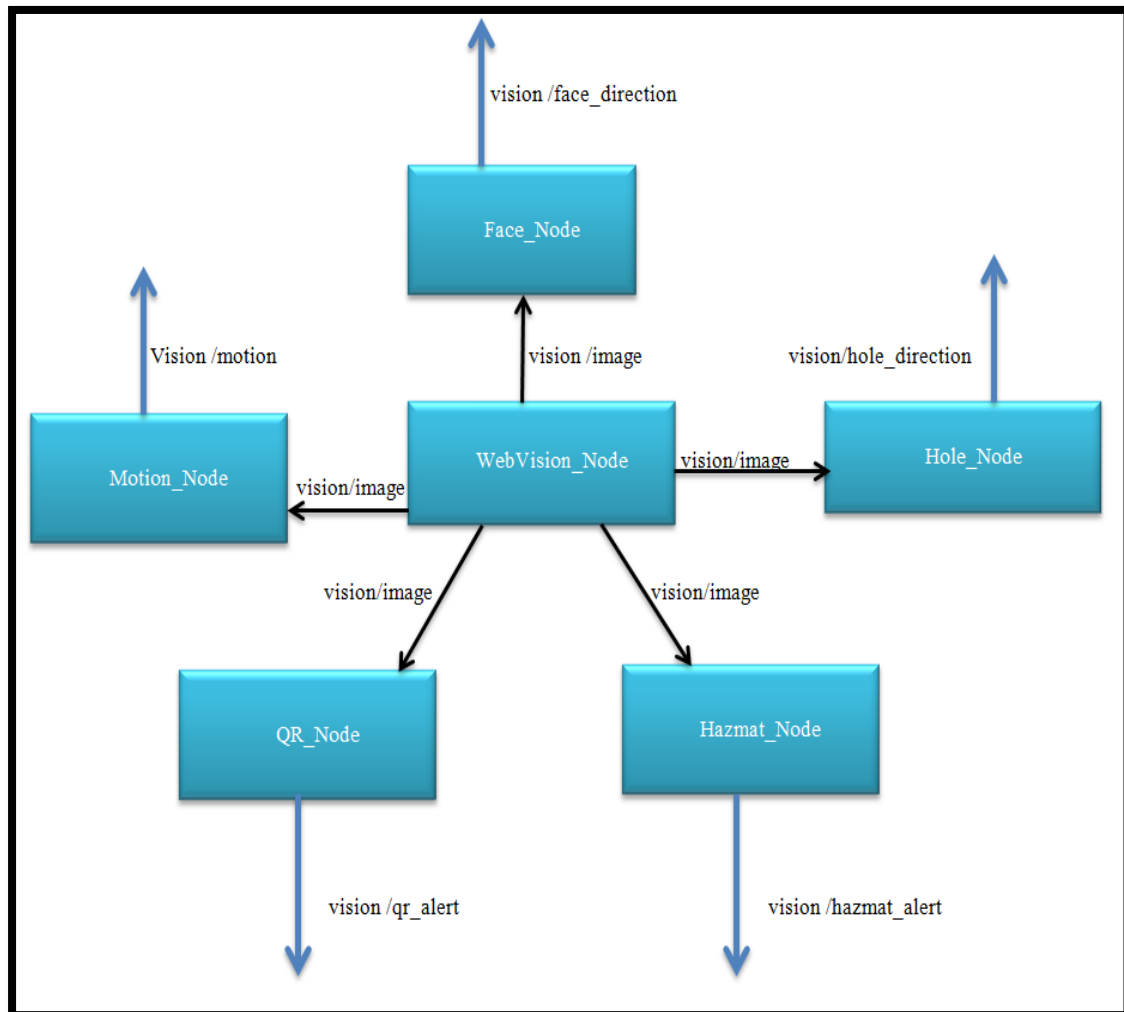


PANDORA VISION DOCUMENTATION

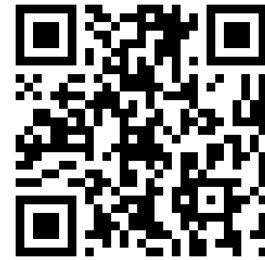
Το vision διαθέτει τους κόμβους που παρουσιάζονται στο ακόλουθο σχήμα. Συγκεκριμένα υπάρχει ένας κοινός κόμβος που αναλαμβάνει να διαθέσει στο ROS κάθε καινούργια εικόνα υπό μορφή μηνυμάτων, ενώ οι υπόλοιποι κόμβοι «ακούνε» αυτά τα μηνύματα και τα επεξεργάζονται κατάλληλα σύμφωνα με τη λειτουργικότητα τους.



Πιο αναλυτικά ο κόμβος `webVision`, είναι υπεύθυνος για την αποστολή εκείνων των μηνυμάτων που περιέχουν τις εικόνες που παίρνουμε από την κάμερα σε όλους τους υπόλοιπους κόμβους. Αξίζει να σημειωθεί ότι ο συγκεκριμένος κόμβος εκπέμπει σχεδόν σύγχρονα με την ταχύτητα λήψης της κάμερας, κάτι το οποίο δεν ισχύει απαραίτητα για όλα τα υπόλοιπα nodes του vision (πχ `hole_node`). Όλοι οι υπόλοιποι κόμβοι εκτός του `webVision`, αναμένουν να τους έρθει μήνυμα με νέα εικόνα. Όταν έρθει η εικόνα εκτελούν όλες τις διαδικασίες που προδιαγράφονται από τη λειτουργικότητα του καθενός και στο τέλος ποστάρουν και αυτοί με τη σειρά τους κατάλληλο μήνυμα.

1. QrCode Node:

Στόχος του συγκεκριμένου κόμβου είναι να αναγνωρίζει την θέση των Qr καθώς και την πληροφορία που αυτά κωδικοποιούν. Οι κώδικες Qr αποτελούν ένα τρόπο κωδικοποίησης ψηφιακής πληροφορίας με οπτικό τρόπο. Ουσιαστικά δουλεύουν όπως οι barcodes, απλά επιτρέπουν την κωδικοποίηση μεγαλύτερης ποσότητας πληροφορίας.



Στην υπάρχουσα υλοποίηση μας η αναγνώριση των QrCodes γίνεται με τη χρήση του `ar`, `zlib`. Το συγκεκριμένο `ar` επιτρέπει την ανάγνωση διαφόρων μορφών κωδίκων και την επιστροφή τους υπό μορφή `string`. Εκτός από το περιεχόμενο, επιστρέφεται το είδος αλλά και τα όρια του κώδικα, πράγμα που μας επιτρέπει να γνωρίζουμε τη θέση του στο `frame`.

Η βιβλιοθήκη `zlib` επιτρέπει την ανάγνωση Qrcodes σε ποικίλα μεγέθη. Για να γίνει όμως αυτή η ανάγνωση, πρέπει τα επιμέρους ψηφία του κώδικα να είναι ευδιάκριτα. Επομένως η λειτουργία της βιβλιοθήκης μπορεί να βελτιωθεί με τη χρήση κάμερας υψηλότερης ανάλυσης και εφαρμογής `sharpening` στην εικόνα. Επιπλέον, ο κώδικας είναι ασπρόμαυρος, οπότε μπορούν να εφαρμοστούν ορισμένα φίλτρα πάνω στην έγχρωμη εικόνα, ώστε να μειώσουν το θόρυβο.

2. Hazmat Node:

Τα hazmats είναι διάφορα σήματα τα οποία μας προειδοποιούν για συγκεκριμένες επικίνδυνες καταστάσεις, όπως αέρια, διαβρωτικά, εύφλεκτα, δηλητήρια, τοξικά, ραδιενεργά, εκρηκτικά κ.α. υλικά. Στη κατηγορία των hazmats τοποθετούμε και τα E charts που χρησιμοποιούν οι οφθαλμίατροι, αρκεί να χρειάζεται μόνο η αναγνώρισή τους.



Παρατηρούμε ότι τα σήματα αυτά διαφέρουν σε χρώμα και στα σχήματα που περιέχουν, ενώ όλα είναι τετράγωνα. Η διαδικασία που ακολουθείται για την αναγνώριση των hazmats είναι η ακόλουθη:

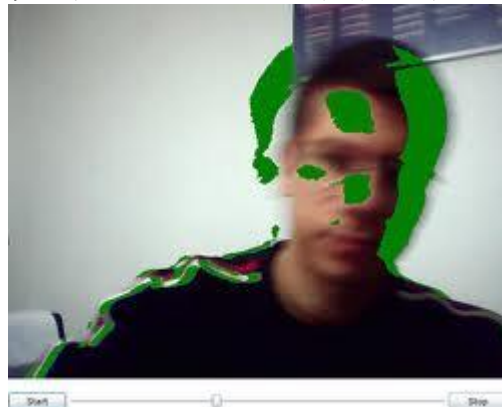
- Αρχικά εκπαιδεύουμε το σύστημα μας να αναγνωρίζει τα συγκεκριμένα σύμβολα, εξάγοντας συγκεκριμένα `features` που έχουμε ορίσει. Είναι σημαντικό σε αυτό το σημείο να προσθέσουμε ότι σύμφωνα με τη λογική που ακολουθούμε **το σύστημα μας είναι σε θέση να εντοπίσει μόνο εκείνα τα hazmats για τα οποία έχει εκπαιδευτεί.**
- Στη συνέχεια αναζητούμε στην εικόνα τετράγωνα σχήματα. Με τη συγκεκριμένη επιλογή δεν κινδυνεύουμε να έχουμε απώλεια πληροφορίας, δεδομένου ότι όλα τα hazmats είναι τετράγωνα.

- Τέλος ακολουθεί σύγκριση των features που βρίσκει ο αλγόριθμος με εκείνα που έχει ήδη αναγνωρίσει κατά τη διαδικασία της εκπαίδευσης. Σε περίπτωση που εντοπίσει μεγάλη ταύτιση ποστάρει στο κατάλληλο topic τη θέση και το id του αντίστοιχου hazard.

Πιο αναλυτικά για να μπορέσουμε να βρούμε τα σημεία ενδιαφέροντος τόσο κατά τη διάρκεια εκπαίδευσης όσο και κατά τη διάρκεια της αναγνώρισης, χρησιμοποιούμε τον αλγόριθμο SURF. Σε αυτό το σημείο είναι σημαντικό να προσθέσω ότι τα χαρακτηριστικά που εντοπίζει είναι ανεξάρτητα κατεύθυνσης καθώς ενδέχεται τα hazard να είναι τοποθετημένα υπό οποιαδήποτε γωνία.

3. Motion Node:

Έστω ότι έχουμε ένα σημείο στο χώρο, με δεδομένη φωτεινότητα. Αν υποθέσουμε ότι ο φωτισμός παραμένει σχετικά σταθερός τότε αν το συγκεκριμένο σημείο παραμείνει ακλόνητο θα προβάλλεται στο ίδιο pixel στη κάμερα, οπότε η απόλυτη αφαίρεση 2 διαδοχικών frame θα δώσει μηδενική τιμή. Αν όμως αυτό το σημείο κινείται, τότε η προβολή του θα είναι διαφορετική στα διαδοχικά frame, με αποτέλεσμα η απόλυτη διαφορά να δίνει τιμή μεγαλύτερη του μηδενός. Βέβαια, υπάρχει η περίπτωση το σημείο να κινείται αλλά η απόλυτη διαφορά να είναι μηδενική, κάτι που συχνά συμβαίνει λόγω της ανικανότητας της κάμερας να ξεχωρίσει σημεία που φέρουν ίδια υφή (texture). Από τα παραπάνω γίνεται εύκολα αντιληπτό ότι η αναγνώριση κίνησης είναι ιδιαίτερα απλή διαδικασία και συνοψίζεται στα ακόλουθα βήματα:



- Αρχικά παίρνουμε μια πρώτη εικόνα από τη κάμερα.
- Μετά από λίγα δευτερόλεπτα παίρνουμε και μία δεύτερη εικόνα.
- Στη συνέχεια εκτελείται απόλυτη αφαίρεση ανάμεσα στις 2 εικόνες και το αποτέλεσμα που προκύπτει είναι μια εικόνα κίνησης. Στη συνέχεια η συγκεκριμένη εικόνα ενισχύεται και φιλτράρεται και αναλόγως διαχωρίζονται 3 περιπτώσεις με αντίστοιχες πιθανότητες η κάθε μια από αυτές:
 - Καθόλου κίνηση ->0% πιθανότητα
 - Μέτρια κίνησης ->50% πιθανότητα
 - Πολλή κίνηση ->100% πιθανότητα

Από όλα τα παραπάνω γίνεται εύκολα αντιληπτό ότι η ευαισθησία του αλγορίθμου μπορεί να επηρεαστεί σε ένα πρώτο επίπεδο από την κατάλληλη τροποποίηση των ορίων των παραπάνω πιθανοτήτων.

4. Face Node:

Ο συγκεκριμένος κόμβος είναι υπεύθυνος για την αναγνώριση θυμάτων βάσει χαρακτηριστικών προσώπου και δέρματος. Η αναγνώριση προσώπου γίνεται με τη μέθοδο του cascade classifier, χρησιμοποιώντας haar-like features. Για την αναγνώριση δέρματος, από τη άλλη συγκρίνουμε κάθε pixel του frame με κατάλληλο ιστόγραμμα που έχει προκύψει από εικόνες δέρματος.



Σε πρώτο στάδιο είναι αναγκαίο να εκπαιδεύσουμε το σύστημα μας έτσι ώστε να είναι σε θέση να αναγνωρίζει πρόσωπα. Αφού έχουμε δημιουργήσει ένα dataset από αρνητικές και θετικές εικόνες εκπαιδεύουμε το σύστημα πάνω σε αυτές, δημιουργώντας ένα δέντρο ταξινομητών. Αν και η εκπαίδευση αποτελεί γενικά μια αυτοματοποιημένη διαδικασία είναι ιδιαίτερα χρονοβόρα, για αυτό επιλέγουμε την νεότερη υλοποίηση της opencv (opencv_traincascade) για εκπαίδευση του cascade classifier, η οποία υλοποιείται παράλληλα.

Συνοπτικά τα βήματα για την αναγνώριση προσώπου σε κάθε frame είναι τα ακόλουθα. Αρχικά κάθε εικόνα που παίρνουμε από την camera περιστρέφεται κατά 5 διαφορετικές γωνίες (0, 45, 315, 90, 270) δημιουργώντας έτσι 5 νέες περιστρεφμένες εικόνες. Ακολούθως δημιουργούνται 5 threads κάθε ένα από τα οποία αναζητά πρόσωπα σε μια από τις παραπάνω εικόνες. Συγκεκριμένα για την αναγνώριση προσώπων χρησιμοποιείται ο Fisherfaces αλγόριθμος. Σε εκείνες τις περιπτώσεις που έχουν αναγνωριστεί πρόσωπα στην παρούσα εικόνα πραγματοποιείται και εξέταση για ύπαρξη δέρματος. Είναι σημαντικό σε αυτό το σημείο να υπογραμμιστεί ότι δεδομένου ότι η υφή του ξύλου μοιάζει πολύ με την υφή του δέρματος συχνά είχαμε προβλήματα κατά την αναγνώριση δέρματος.

5. Hole Detection Node:

Ο συγκεκριμένος κόμβος όπως γίνεται εύκολα κατανοητό από το όνομα του είναι υπεύθυνος για τον εντοπισμό τρύπας. Το πρώτο βήμα είναι να αναγνωρίσει το σύστημα τι ανήκει στην έννοια «τοίχος» και τι όχι. Στο διαγωνισμό οι τοίχοι είναι φτιαγμένοι από ξύλο και έχουν συγκεκριμένη υφή. Εκτελώντας λοιπόν επαναπροβολή (back projection) μπορούμε να εκτιμήσουμε ποιες περιοχές της τρέχουσας εικόνας αντιστοιχούν πιθανότερα σε ξύλινες επιφάνειες. Συνδυάζοντας αυτή τη πληροφορία με template matching διαφόρων δειγμάτων ξύλου, έχουμε μια μάσκα που μας πληροφορεί σχετικά με την

ύπαρξη ή όχι ξύλου. Μιας και οι τοίχοι δεν έχουν μόνο ξύλο, αλλά και άλλα υλικά (πχ το μπλε βαρέλι), εκπαιδεύουμε το σύστημα ώστε να θεωρεί και τα λοιπά υλικά τοίχο.

Στη συνέχεια προσπαθούμε να εντοπίσουμε περιοχές που βρίσκονται πλήρως μέσα σε τοίχο, οι οποίες δεν ανήκουν σε αυτόν. Τέτοιες περιοχές είναι φυσικά οι τρύπες, οι οποίες είτε είναι πολύ σκούρες, είτε φαίνονται μέσα από αυτές διάφορα αντικείμενα, τα οποία δε θεωρούνται τοίχος. Επειδή όμως το ταβάνι και το πάτωμα δε θεωρούνται επίσης τοίχος, θα μπορούσε κανείς να θεωρήσει και αυτά τρύπες. Εδώ έρχεται να βοηθήσει ο περιορισμός ότι η τρύπα πρέπει να περιλαμβάνεται εξολοκλήρου στο frame της κάμερας και να περικλείεται από τοίχο.



Με τον εντοπισμό της τρύπας αναλαμβάνει δράση ο tracker. Ο tracker παρακολουθεί τη τρύπα καθώς αυτή μετακινείται μέσα στο frame της κάμερας. Γνωρίζοντας τη κίνηση της τρύπας, μπορούμε να προβλέψουμε την επόμενη θέση της. Αν η πρόβλεψή μας ταιριάζει με τη θέση που μετράμε από τη κάμερα, τότε υποθέτουμε ότι όντως πρόκειται για τρύπα και ότι την παρακολουθούμε επιτυχώς. Εφόσον συνεχίζει να υπάρχει αυτή η οντότητα ανεβάζουμε τη πιθανότητά της οπότε αποδίδουμε μια πιθανότητα εύρεσης τρύπας. Ο tracker λειτουργεί και στις περιπτώσεις που η τρύπα χάνεται από το frame ή βρίσκεται στα όριά του, αρκεί αυτός ο χρόνος να μην είναι ιδιαίτερα μεγάλος.

Μπορούμε να φανταστούμε τον tracker σαν έναν μεγάλο πίνακα. Στη μία διάσταση έχουμε τα trackerchains, αλυσίδες δηλαδή από things τα οποία παρακολουθούνται. Αυτές οι αλυσίδες αποδίδονται σε μια φυσική τρύπα και περιέχουν αντίγραφα των things στα διάφορα frames που εντοπίστηκαν. Έτσι μια φυσική τρύπα A θα έχει στο trackerchain της το thing1 από το frame X, το thing2 από το frame Y, το thing3 από το frame Z κτλ. Ο tracker αποφάσισε ότι τα thing1 2 και 3 ανήκουν στην ίδια φυσική τρύπα, οπότε τα τοποθετεί στο ίδιο trackerchain. Η άλλη διάσταση του πίνακα έχει τρύπες που βρέθηκαν στο τρέχον frame και δεν έχουν τοποθετηθεί σε trackerchain. Ο πίνακας συμπληρώνεται με έναν βαθμό ομοιότητας, ο οποίος βασίζεται στην ομοιότητα σχήματος, μεγέθους, απόστασης και θέσης της εν λόγω τρύπας με κάθε εκπρόσωπο των trackerchains. Η τρύπα θα αντιστοιχηθεί με το trackerchain που φέρει το μεγαλύτερο βαθμό. Αν ο βαθμός αυτός είναι κατώτερος ενός ορίου, τότε δημιουργείται ένα νέο trackerchain, καθώς θεωρούμε ότι πρόκειται μάλλον για νέα τρύπα. Απ' όσα έχουμε αναφέρει μέχρι τώρα, καταλαβαίνουμε ότι αυτός ο trackerchain θα έχει χαμηλή πιθανότητα. Αν οι trackerchains δεν αποκτήσουν νέο μέλος (νέο thing) μέσα σε ένα εύλογο χρονικό διάστημα τότε θεωρούνται ανενεργοί και διαγράφονται. Αυτό βοηθάει ώστε να μην συσσωρεύονται τρύπες που ανήκουν σε παλιά frames.

6. Stereo_ps3 Node:

Ο συγκεκριμένος κόμβος αποτελεί ένα ξεχωριστό κομμάτι του vision και δεν επικοινωνεί με τα υπόλοιπα υποσυστήματα. Ωστόσο για λόγους πληρότητας παραθέτω περιληπτικά την λειτουργία του. Η λογική που επιλέξαμε να ακολουθήσουμε για τον εντοπισμό τρύπας παρουσιάζει σημαντικές ομοιότητες με την λογική στην περίπτωση της monocular camera. Συγκεκριμένα σε πρώτο στάδιο προσπαθούμε να εντοπίσουμε περιοχές ενδιαφέροντος στην εικόνα που βλέπουν οι δυο κάμερας και στη συνέχεια συμπεραίνουμε αν είναι τρύπα ή απλά κυκλοειδής επιφάνεια χρησιμοποιώντας την πληροφορία του βάθους που μας παρέχει η στερεοσκοπική κάμερα.

Συνοπτικά τα βήματα της υλοποίησης μας είναι τα ακόλουθα:

- Είναι γνωστό ότι πιθανές θέσεις τρύπας είναι πάνω σε ξύλα. Συνεπώς μπορούμε να περιορίσουμε τις περιοχές ενδιαφέροντος της σκηνής μας σε εκείνες που βρίσκονται πάνω σε ξύλα και σε όλες τις υπόλοιπες. Η συγκεκριμένη παραδοχή είναι πολύ σημαντική καθώς περιορίζει σημαντικά το μέγεθος της περιοχής που ψάχνουμε κάθε φορά.
- Σε δεύτερο στάδιο εφαρμόζεται στην εικόνα μας κατάλληλο φίλτρο, το οποίο εντοπίζει όλες τις ακμές στις περιοχές που μας ενδιαφέρουν. Σε αυτό το σημείο και εφόσον έχουμε τις ακμές ψάχνουμε να βρούμε κυκλικές περιοχές που εμφανίζονται στην παρούσα εικόνα.
- Μέχρι αυτό το σημείο, δεν έχουμε καμία πληροφορία για το αν η κυκλική περιοχή που εντοπίσαμε στα προηγούμενα βήματα είναι όντως τρύπα. Για να μπορέσουμε επομένως να διαχωρίσουμε τις τρύπες από τα πιθανά false alarms χρησιμοποιούμε σε αυτό το σημείο την πληροφορία του βάθους που έχουμε εξασφαλίσει από την στερεοσκοπική κάμερα. Έτσι για κάθε μια από τις περιοχές ενδιαφέροντος (κυκλοειδείς επιφάνειες-blobs) συγκρίνουμε βάθη και ανάλογα καταλήγουμε στο αν είναι τρύπα ή όχι.
- Στην πορεία της υλοποίησης μας παρατηρήσαμε ότι η πληροφορία του βάθους που εξασφαλίζουμε σε μερικές περιπτώσεις έχει μεγάλα σφάλματα επομένως αποφασίσαμε να εισάγουμε μια επιπλέον δικλίδα ασφαλείας για να μειώσουμε την πιθανότητα σφάλματος. Έτσι για να καταλήξουμε ότι μια τρύπα είναι όντως τρύπα και όχι false alarm είναι αναγκαίο να εντοπιστεί σε 5 συνεχόμενα frames.
- Τέλος για να γνωρίζουμε ανά πάσα στιγμή τη θέση της τρύπας, που έχουμε εντοπίσει, την κάνουμε track όσο αυτή εμφανίζεται στις εικόνες που λαμβάνουμε από τις κάμερες.