



ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΑΥΤΟΜΑΤΙΣΜΟΥ Τ.Ε.
ΑΤΕΙ ΘΕΣΣΑΛΟΝΙΚΗΣ

**ΕΠΕΞΕΡΓΑΣΙΑ ΕΙΚΟΝΑΣ ΓΙΑ
ΑΥΤΟΜΑΤΟΠΟΙΗΜΕΝΟ ΜΗ ΚΑΤΑΣΤΡΟΦΙΚΟ
ΕΛΕΓΧΟ ΥΓΕΙΑΣ ΥΠΟΔΟΜΩΝ**

ΧΡΙΣΤΟΠΟΥΛΟΣ ΒΑΣΙΛΕΙΟΣ

Επιβλέποντες καθηγητές: Ξανθός Στέλιος, Παπαχριστοφόρου Μιχάλης

Περίληψη

Οι ρωγμές από σκυρόδεμα είναι ένα φαινόμενο, το οποίο υπάρχει σε όλους τους τύπους κατασκευών από σκυρόδεμα. Η παρακολούθησή τους είναι επιθυμητή, αφού υποδεικνύουν το πρόβλημα που υπάρχει. Ο προσδιορισμός και η παρακολούθηση της ποσότητας και της σοβαρότητας της ρωγμής, είναι πρωταρχικής σημασίας για την αξιολόγηση της τρέχουσας κατάστασης και πρόβλεψη της μελλοντικής διάρκειας ζωής μιας κατασκευής. Οι ρωγμές από σκυρόδεμα μπορεί να υποδεικνύουν τη διάβρωση της ενίσχυσης, την ανάπτυξη θραυσμάτων ή την αλλαγή των συνθηκών στήριξης. Επομένως, η παρακολούθηση των ρωγμών κατά την διάρκεια ζωής των κατασκευών από σκυρόδεμα αποτελεί αποτελεσματική τεχνική για την αξιολόγηση του επιπέδου ασφαλείας και την προετοιμασία σχεδίων για μελλοντική κατάλληλη αποκατάσταση.

Μια αναπτυσσόμενη τεχνική είναι οι μη επανδρωμένες επιθεωρήσεις που χρησιμοποιούν μη επανδρωμένα αεροσκάφη (Unmanned Aerial Vehicles) (UAV). Τα UAVs είναι αεροσκάφη εξοπλισμένα με κάμερες, αισθητήρες, GPS, κλπ. Οι εικόνες RGB (έγχρωμες εικόνες σε red, green and blue color space) λαμβάνονται από μια κάμερα που είναι τοποθετημένη σε UAV που πετάει γύρο από την κατασκευή, για την ανίχνευση ρωγμών και άλλων ελαττωμάτων. Κάθε εικόνα που έχει ληφθεί από το UAV πρέπει να αξιολογηθεί για τον σχηματισμό ρωγμών. Για την εξοικονόμηση χρόνου, αυτή η εργασία μπορεί να γίνει εφαρμόζοντας τεχνικές επεξεργασίας εικόνας για την αυτόματη ανίχνευση ρωγμών και ελαττωμάτων, αντί η χρήση ανθρώπου για την αναγνώριση των ελαττωμάτων.

Η εργασία αποτελείται από τα ακόλουθα τμήματα: Μια εισαγωγή και ανάλυση του τρόπου με τον οποίο ο υπολογιστής αντιλαμβάνεται και επεξεργάζεται τις εικόνες, κεφάλαιο 1. Η προτεινόμενη μέθοδος για την αυτόματη ανίχνευση ρωγμών, παρουσιάζεται στο κεφάλαιο 2. Τα πειραματικά αποτελέσματα της προτεινόμενης μεθόδου παρουσιάζονται στο κεφάλαιο 3. Τα συμπεράσματα και οι μελλοντική εργασία παρουσιάζονται στο κεφάλαιο 4. Τέλος στο Παράρτημα παρουσιάζεται ο κώδικας Matlab.

Abstract

Cracks in concrete cement exist in all constructions. Their development is commonly desirable since they indicate an existing problem. Determining the quantity of cracks and surveillancing the severity of them is of outmost importance regarding the current condition of the concrete and for determining the lifespan of a construction. Cracks in the concrete may be an indication of corrosion in the metallic armoring, shard development, or alteration in the shuttering conditions. Thus, the surveillance of cracks throughout the lifespan of concrete-based constructions is an efficient technique to evaluate the safety level of the construction itself and for planning future restoration/repair.

A newly introduced technique for surveillance is the use of unmanned aerial vehicles (UAV). UAVs are drones equipped with cameras, sensors, GPS, etc. RGB pictures (colored pictures in red, green, and blue tones) are captured by cameras loaded in the UAV, which flies around the construction, seeking to locate cracks and other defects. Each picture taken by the UAV should be evaluated to located cracks. For time-saving purposes, the evaluation can be done by using image processing techniques for auto-detecting cracks and defects, rather than using human optical inspection to locate them.

The current work is organized as follows: An introduction on how computers understand and process images is presented in chapter 1. The proposed method for auto-detecting cracks is introduced in chapter 2. Experimental results of the proposed method are shown in chapter 3. Future expansions and inferences regarding the current work are concluded in chapter 4. Finally, in the appendix, it is presented the Matlab code used.

Ευχαριστίες

Στο σημείο αυτό θα ήθελα να ευχαριστήσω τους επιβλέποντες καθηγητές μου Δρ. Στέλιο Ξανθό και Δρ. Μιχάλη Παπαχριστοφόρου, τόσο για την ανάθεση της πτυχιακής εργασίας, όσο και για την βοήθεια που μου προσέφεραν και την άψογη συνεργασία που είχα μαζί τους. Επιπλέον θέλω να ευχαριστήσω ιδιαίτερω την οικογένειά μου, για την υπομονή και την στήριξη που μου προσέφεραν όλο αυτό το χρονικό διάστημα.

Περιεχόμενα

Περίληψη	3
Abstract	5
Ευχαριστίες	7
Ευρετήριο εικόνων	10
1 Εισαγωγή	12
1.1 Ανάκτηση εικόνας	13
1.2 Τύποι χρωματικών μοντέλων και color space	14
2 Μέθοδος ελέγχου υγείας υποδομών	17
2.1 Βήματα της μεθόδου ελέγχου υγείας υποδομών	17
2.2 Λεπτομερής επεξήγηση των βημάτων της μεθόδου	18
3 Πειραματικά αποτελέσματα	27
3.1 Πειραματικά αποτελέσματα της προτεινόμενης μεθόδου	27
4 Συμπεράσματα και μελλοντική εργασία	52
4.1 Συμπεράσματα προτεινόμενης μεθόδου	52
4.2 Μελλοντική εργασία	53
Παράρτημα: Κώδικας Matlab	55
Βιβλιογραφία	57

Ευρετήριο εικόνων

Εικόνα 1. Εικόνα από UAV	13
Εικόνα 2. Τα τρία χρωματικά στοιχεία του πίνακα	14
Εικόνα 3. RGB (Red, Green and Blue) color space	15
Εικόνα 4. CMYK (Cyan, Magenta, Yellow and Black)	15
Εικόνα 5. YUV color space	16
Εικόνα 6. Μέθοδος Otsu	20
Εικόνα 7. Αναπαράσταση γραμμής σε κανονική μορφή	24
Εικόνα 8. Συντελεστής κυκλικότητας	25
Εικόνα 9. Τοίχος κτηρίου Αυτοματισμού 1m απόσταση	28
Εικόνα 10. Τοίχος κτηρίου Αυτοματισμού 0.75m απόσταση	28
Εικόνα 11. Τοίχος κτηρίου Αυτοματισμού 0.5m απόσταση	29
Εικόνα 12. Τοίχος κτηρίου Αυτοματισμού 0.25m απόσταση	29
Εικόνα 13. Επεξεργασμένη εικόνα 9	30
Εικόνα 14. Επεξεργασμένη εικόνα 10	30
Εικόνα 15. Επεξεργασμένη εικόνα 11	31
Εικόνα 16. Επεξεργασμένη εικόνα 12	31
Εικόνα 17. Τοίχος κτηρίου Διοίκησης 1m απόσταση	32
Εικόνα 18. Τοίχος κτηρίου Διοίκησης 0.75m απόσταση	33
Εικόνα 19. Τοίχος κτηρίου Διοίκησης 0.5m απόσταση	33
Εικόνα 20. Τοίχος κτηρίου Διοίκησης 0.25m απόσταση	34
Εικόνα 21. Επεξεργασμένη εικόνα 17	34
Εικόνα 22. Επεξεργασμένη εικόνα 18	35
Εικόνα 23. Επεξεργασμένη εικόνα 19	35
Εικόνα 24. Επεξεργασμένη εικόνα 20	36
Εικόνα 25. Τοίχος κτηρίου Οχημάτων 1m απόσταση	37
Εικόνα 26. Τοίχος κτηρίου Οχημάτων 0,75m απόσταση	38
Εικόνα 27. Τοίχος κτηρίου Οχημάτων 0.5m απόσταση	38
Εικόνα 28. Τοίχος κτηρίου Οχημάτων 0.25m απόσταση	39
Εικόνα 29. Τοίχος κτηρίου Οχημάτων 0.2m απόσταση	39
Εικόνα 30. Επεξεργασμένη εικόνα 25	40
Εικόνα 31. Επεξεργασμένη εικόνα 26	40
Εικόνα 32. Επεξεργασμένη εικόνα 27	41
Εικόνα 33. Επεξεργασμένη εικόνα 28	41
Εικόνα 34. Επεξεργασμένη εικόνα 29	42
Εικόνα 35. Τοίχος κτηρίου Οχημάτων 1m απόσταση	43
Εικόνα 36. Τοίχος κτηρίου Οχημάτων 0.75m απόσταση	43
Εικόνα 37. Τοίχος κτηρίου Οχημάτων 0.5m απόσταση	44
Εικόνα 38. Τοίχος κτηρίου Οχημάτων 0.25m απόσταση	44
Εικόνα 39. Τοίχος κτηρίου Οχημάτων 0.2m απόσταση	45
Εικόνα 40. Επεξεργασμένη εικόνα 35	46
Εικόνα 41. Επεξεργασμένη εικόνα 36	46
Εικόνα 42. Επεξεργασμένη εικόνα 37	47
Εικόνα 43. Επεξεργασμένη εικόνα 38	47
Εικόνα 44. Επεξεργασμένη εικόνα 39	48
Εικόνα 45. Πλάκες πεζοδρομίου 1m απόσταση	49

Εικόνα 46. Επεξεργασμένη εικόνα 45	49
Εικόνα 47. Άσφαλτος 1m απόσταση	50
Εικόνα 48. Επεξεργασμένη εικόνα 47	51

Κεφάλαιο 1

1 Εισαγωγή

Οι ρωγμές στο σκυρόδεμα είναι ένα φαινόμενο, το οποίο υπάρχει σε όλους τους τύπους κατασκευών από σκυρόδεμα. Ο προσδιορισμός και η παρακολούθηση της ποσότητας και της σοβαρότητας της ρωγμής είναι πρωταρχικής σημασίας για την αξιολόγηση της τρέχουσας κατάστασης και την πρόβλεψη της μελλοντικής διάρκειας ζωής μιας κατασκευής. Οι ρωγμές από σκυρόδεμα μπορεί να υποδεικνύουν τον ανεπαρκή οπλισμό, τη διάβρωση του οπλισμού, την καθίζηση ή την υπέρβαση του ορίου αντοχής της κατασκευής. Επομένως, η παρακολούθηση των ρωγμών κατά την διάρκεια ζωής των κατασκευών από σκυρόδεμα αποτελεί αποτελεσματική τεχνική για την αξιολόγηση του επιπέδου ασφαλείας και την προετοιμασία σχεδίων για μελλοντική κατάλληλη αποκατάσταση.

Ρωγμή ορίζεται ως η ασυνέχεια του υλικού, η οποία ασυνέχεια έχει τη μία διάσταση πολύ μικρότερη από τις άλλες. Οι ρωγμές στις κατασκευές από σκυρόδεμα είναι κάτι ιδιαίτερα συνηθισμένο, αφού υποδηλώνουν την ανάπτυξη παραμορφώσεων στο φορέα. Η παρακολούθησή τους είναι επιθυμητή, αφού υποδεικνύουν το πρόβλημα που υπάρχει. Σκοπός λοιπόν είναι η εύρεση του σημείου της υποδομής που υπάρχει η ρωγμή, ώστε να αντιμετωπιστεί έγκαιρα το πρόβλημα.

Η χρήση του ανθρώπινου οπτικού ελέγχου είναι η παλαιότερη και πιο αξιόπιστη μέθοδος αναγνώρισης ρωγμών σε κατασκευές από σκυρόδεμα. Ωστόσο, η χρήση ανθρώπινων επιθεωρητών είναι χρονοβόρα, δαπανηρή και μπορεί να θέσει σε κίνδυνο την ανθρώπινη ασφάλεια. Η ρουτίνα επιθεώρησης απαιτεί επίσης, τη διακοπή της κυκλοφορίας, για παράδειγμα, σε μια γέφυρα, σε ένα δρόμο ή σε μια σήραγγα, η χρήση ειδικού εξοπλισμού, για παράδειγμα σε κτήρια, ώστε να γίνει η επιθεώρηση σωστά και με ασφάλεια. Έτσι, οι μη επανδρωμένες μέθοδοι επιθεώρησης αυξάνονται τα τελευταία χρόνια προκειμένου να μειωθεί ο χρόνος και η δαπάνη της επιθεώρησης. Μια από τις τεχνικές μη επανδρωμένης επιθεώρησης είναι η χρήση Unmanned Aerial Vehicles (UAV) (Μη επανδρωμένα αεροσκάφη). Τα UAVs είναι αεροσκάφη εξοπλισμένα με κάμερες, αισθητήρες, GPS, κλπ. Οι εικόνες RGB (έγχρωμες εικόνες σε red, green and blue color space) λαμβάνονται από μια κάμερα που είναι τοποθετημένη σε UAV που πετάει γύρω από την κατασκευή, για την ανίχνευση ρωγμών και άλλων ελαττωμάτων (Εικόνα 1).



Εικόνα 1. Εικόνα από UAV

Κάθε εικόνα που έχει ληφθεί από το UAV πρέπει να αξιολογηθεί για τον σχηματισμό ρωγμών. Για την εξοικονόμηση χρόνου, αυτή η εργασία μπορεί να γίνει εφαρμόζοντας τεχνικές επεξεργασίας εικόνων για την αυτόματη ανίχνευση ρωγμών και ελαττωμάτων, αντί της χρήσης ανθρώπου για την αναγνώριση των ελαττωμάτων.

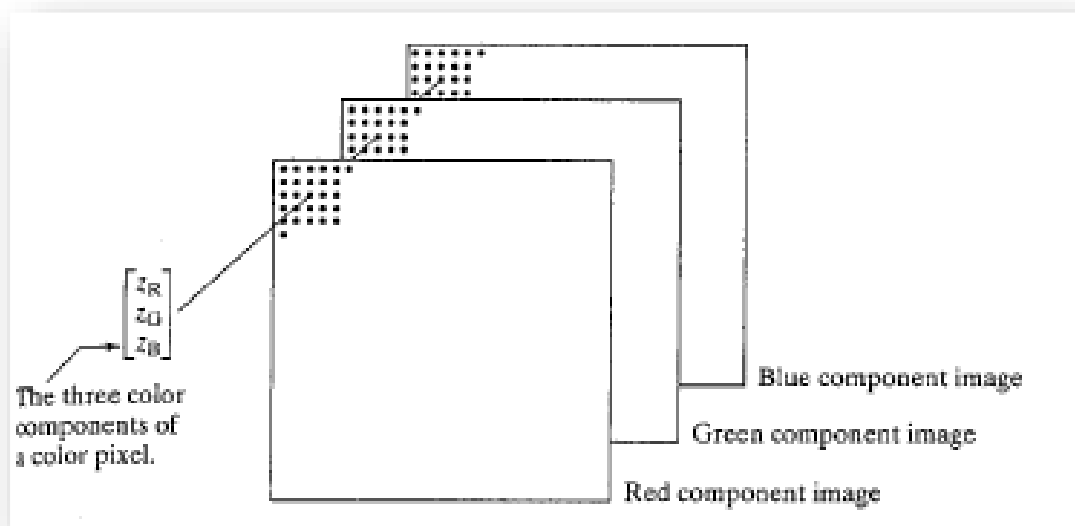
Η παρούσα εργασία βασίστηκε στο έργο των Sattar Dorafshan, Marc Maguire και Xiaojun Qi “Automatic Surface Crack Detection in Concrete Structures Using OTSU Thresholding and Morphological Operations” (2016)”. Με αρκετές τροποποιήσεις και προσθήκες μορφολογικών λειτουργιών και μεθόδων διαχωρισμού των ρωγμών από άλλα αντικείμενα. Εκτός των τροποποιήσεων στον αλγόριθμο, προστεθήκανε και μέθοδοι για την εξάλειψη και άλλων ελαττωμάτων από το τελικό αποτέλεσμα. Η αφαίρεση των απόλυτων ευθειών, όπως μια δοκός ή ο σπλισμός της υποδομής που μπορεί να διακρίνεται σε κάποια σημεία, η αφαίρεση των αντικειμένων που δεν μοιάζουν με ευθεία, μέθοδος κυκλικότητας, όπου γίνεται και η τελική διάγνωση των αποτελεσμάτων της επεξεργασίας των εικόνων, είναι μερικές από τις μεθόδους. Σε αυτή την τελική διάγνωση το πρόγραμμα επεξεργάζεται για τελευταία φορά τις εικόνες, και απορρίπτει όποια ελαττώματα έχουν βρεθεί μέχρι εκείνη στιγμή και δεν μοιάζει με ρωγμή. Έτσι στις τελικές εικόνες, μετά την επεξεργασία των αρχικών εικόνων, θα έχουν παραμείνει μόνο οι ρωγμές.

1.1 Ανάκτηση εικόνας

Στην παρούσα εργασία δεν χρησιμοποιήθηκε κάποιο UAV για την λήψη των εικόνων, αλλά μια απλή ψηφιακή φωτογραφική μηχανή με ανάλυση 20-megapixels.

Αυτό έγινε για καθαρά οικονομικούς λόγους. Πριν αναλύσουμε όμως την διαδικασία επεξεργασίας των εικόνων θα πρέπει να κάνουμε μια αναφορά στο πως αντιλαμβάνεται και διαχειρίζεται ο Η/Υ την εικόνα και με πια ψηφιακή μορφή την αποδίδει στον χρήστη.

Ο Η/Υ αντιλαμβάνεται την εικόνα ως μια ακολουθία πλήθους στιγμιότυπων ανά δευτερόλεπτο (frame per second), τα οποία αποθηκεύει ή διαχειρίζεται υπό την μορφή διδιάστατων πινάκων ή σύνολο πινάκων. Το κάθε κελί του πίνακα αντιστοιχεί σε ένα pixel και περιέχει την χρωματική τιμή του.



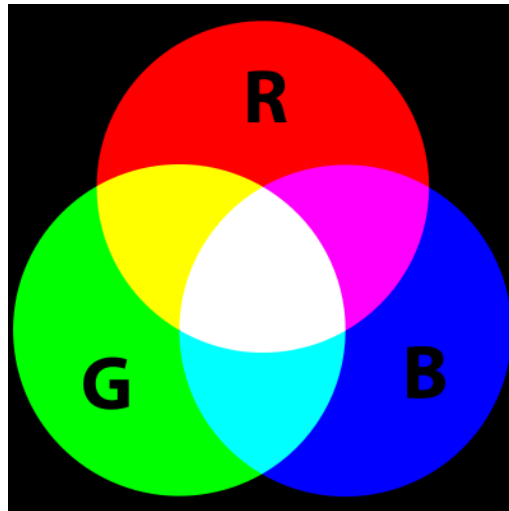
Εικόνα 2. Τα τρία χρωματικά στοιχεία του πίνακα

Η μορφή της χρωματικής τιμής του κάθε κελιού, καθώς και το αν κάθε στιγμιότυπο αντιστοιχεί σε έναν ή σε σύνολο πινάκων, καθορίζεται από το χρωματικό μοντέλο (color space) που χρησιμοποιεί η κάμερα για να χαρτογραφήσει το αρχείο που θα περάσουμε στον υπολογιστή.

1.2 Τύποι χρωματικών μοντέλων και color space

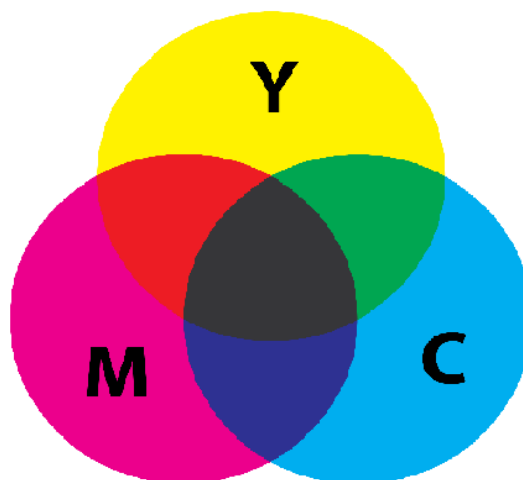
Ένα χρωματικό μοντέλο είναι ένα αφηρημένο μαθηματικό μοντέλο που περιγράφει τον τρόπο που τα χρώματα μπορούν να εκπροσωπούνται ως αριθμητικές τιμές (συνήθως τρεις ή τέσσερις τιμές), ή ως χρωματικά συστατικά (RGB και CMYK είναι χρωματικά μοντέλα). Ωστόσο, ένα μοντέλο χρωμάτων που δεν συσχετίζει την λειτουργία χαρτογράφησης του με έναν απόλυτο χρωματικό χώρο (color space), είναι λίγο ή πολύ ένα αυθαίρετο σύστημα χρωμάτων που δεν σχετίζεται με το παγκοσμίως κατανοητό σύστημα ερμηνείας για το χρώμα.

Οι πιο διαδεδομένοι χρωματικοί χώροι είναι οι RGB, CMYK και YUV. Ο RGB (red, green and blue) χρησιμοποιεί προσθετική χρωματική ανάμειξη (additive color mixing), διότι περιγράφει το είδος του φωτός που πρέπει να εκπέμψουν για να παράγουν ένα συγκεκριμένο χρώμα. Οι τρεις τιμές του φωτός προστίθενται μαζί για να δημιουργήσουν μορφή από το σκοτάδι.



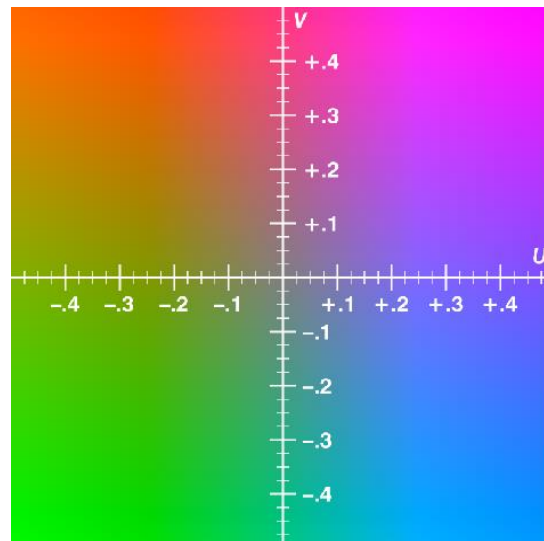
Εικόνα 3. RGB (Red, green and blue) color space

Ο CMYK (cyan, magenta, yellow and black) χρησιμοποιεί αφαιρετική χρωματική ανάμειξη (subtractive color mixing) που χρησιμοποιείται στη διαδικασία εκτύπωσης, επειδή περιγράφει τι είδους μελάνια πρέπει να εφαρμοστούν ώστε το φως που ανακλάται από το υπόστρωμα και μέσω των μελανιών, να παράγει ένα συγκεκριμένο χρώμα. Κάποιος ξεκινάει με ένα λευκό υπόστρωμα (καμβά, σελίδα, κλπ.), και χρησιμοποιεί μελάνι για να αφαιρέσει το χρώμα από το λευκό και να δημιουργήσει μια εικόνα.



Εικόνα 4. CMYK (Cyan, Magenta, Yellow and black) color space

Το μοντέλο YUV καθορίζει έναν χρωματικό χώρο από την άποψη ενός συστατικού φωτεινότητας (Y) και δύο χρωματικά συστατικά (UV). Χρησιμοποιείται συνήθως ως μέρος ενός αγωγού χρωματικής εικόνας, το οποίο κωδικοποιεί μια έγχρωμη εικόνα ή ένα βίντεο λαμβάνοντας την ανθρώπινη αντίληψη υπόψη, επιβάλλοντας μειωμένο εύρος ζώνης για τα χρωματικά συστατικά. Με αυτόν τον τρόπο επιτρέπει να φιλτράρονται πιο αποτελεσματικά τα τυπικά σφάλματα μετάδοσης ή τα συμπιεσμένα αντικείμενα από την ανθρώπινη αντίληψη, απ' ό,τι θα φιλτράρονταν με "απευθείας" χρήση του RGB χρωματικού μοντέλου.



Εικόνα 5. YUV color space

Κεφάλαιο 2

2 Μέθοδος ελέγχου υγείας υποδομών

Σε αυτή την ενότητα θα εξηγηθεί η μέθοδος για τον έλεγχο της υγείας υποδομών. Η προτεινόμενη μέθοδος παρέχει έναν απλό και γρήγορο αλγόριθμο αυτόματης ανίχνευσης ρωγμών σε κατασκευές από σκυρόδεμα με πολλαπλές λειτουργίες φιλτραρίσματος, πολλαπλές λειτουργίες κατωφλιού και διάφορες μορφολογικές λειτουργίες. Επιπλέον, πολλές εμπειρικές στρατηγικές χρησιμοποιήθηκαν στην υλοποίηση του αλγορίθμου, ώστε ο αλγόριθμος να είναι σε θέση να αναγνωρίσει τις εικόνες οι οποίες έχουν ρωγμές από εκείνες που δεν έχουν κανένα ελάττωμα. Επίσης ο αλγόριθμος είναι σε θέση να αναγνωρίζει άλλα ελαττώματα, όπως τις άκρες του σκυροδέματος, τις ενώσεις των πλακών σε πεζοδρόμια, κάποια μογιά που μπορεί να έχει η υποδομή, η βρωμιά στην επιφάνεια, οι σκιές και άλλες ατέλειες που μπορεί να είναι πρόβλημα για την υποδομή αλλά δεν είναι ρωγμές, και να τα αφαιρεί από το τελικό αποτέλεσμα. Έτσι ο αλγόριθμος είναι σε θέση να εντοπίζει μόνο τις ρωγμές, και να παραβλέπει τα υπόλοιπα ελαττώματα που μπορεί να υπάρχουν.

2.1 Βήματα της μεθόδου ελέγχου υγείας υποδομών

Σαν είσοδος του αλγορίθμου είναι οποιαδήποτε εικόνα RGB με επέκταση “jpg” οποιουδήποτε μεγέθους. Ωστόσο, το πρόγραμμα μπορεί εύκολα να τροποποιηθεί για οποιεσδήποτε άλλες επεκτάσεις. Ο αλγόριθμος έχει γραφτεί σε λογισμικό Matlab R2015a. Ο αλγόριθμος που σχεδιάστηκε δεν χρησιμοποιεί μέθοδο εκπαίδευσης, έτσι είναι γρήγορος και αποτελεσματικός.

Για την ανίχνευση ρωγμών στην προτεινόμενη μέθοδο, απαιτούνται τα ακόλουθα βήματα :

1. Εισαγωγή της RGB εικόνας, διαβάζεται από αποθηκευμένες εικόνες στον υπολογιστή
2. Η RGB εικόνα μετατρέπεται σε κλίμακα του γκρι
3. Στην εικόνα της κλίμακας του γκρι εφαρμόζεται ένα μεσαίο φίλτρο, για την εξομάλυνση της επιφάνειας
4. Εφαρμόζεται ανιχνευτής ακμών Sobel για την ενδυνάμωση των άκρων της εικόνας
5. Εφαρμόζεται η μέθοδος κατωφλιού Otsu για την μετατροπή της εικόνας σε ασπρόμαυρη, για την απόκτηση της δυαδικής εικόνας
6. Αναγνώριση μικρών περιοχών που το μέγεθός τους εξαρτάται από το μέγεθος της εικόνας, εδώ εκλαμβάνεται 1000 λόγω των 20Mpixel, και τοποθέτηση αυτών στο παρασκήνιο

7. Αναγνώριση περιοχών με προσανατολισμό 0,90 και -90 μοίρες και τοποθέτηση αυτών στο παρασκήνιο
8. Χρησιμοποιείται η μορφολογική λειτουργία “majority” για τη σύνδεση των κοντινών περιοχών μεταξύ τους
9. Αναγνώριση περιοχών με μέγεθος μικρότερο από 200 pixels, μικρές περιοχές που δημιουργούνται κατά την εκτέλεση της μορφολογικής λειτουργίας του προηγούμενου βήματος, και τοποθέτηση αυτών στο παρασκήνιο
10. Αναγνώριση περιοχών που είναι απόλυτες ευθείες, και αφαίρεση αυτών
11. Αναγνώριση περιοχών που μοιάζουν με κύκλο και αφαίρεση αυτών
12. Αποθήκευση επεξεργασμένων εικόνων στον υπολογιστή

2.2 Λεπτομερής επεξήγηση των βημάτων της μεθόδου

Σε αυτό το κεφάλαιο θα αναλυθούν τα βήματα του κώδικα ξεχωριστά.

Βήμα 1

Το πρώτο βήμα είναι να διαβάσουμε τις εικόνες, που έχουμε αποθηκεύσει στον υπολογιστή, και τις οποίες τις έχουμε πάρει από μια ψηφιακή φωτογραφική μηχανή. Οι εντολές που χρησιμοποιήθηκαν για την υλοποίηση αυτού του βήματος είναι οι εξής:

```
str=['image' num2str(i) '.jpg'];  
orim=imread(str);
```

Βήμα 2

Το δεύτερο, και πολύ σημαντικό βήμα, είναι η μετατροπή της RGB εικόνας, που έχει γίνει λήψη από την φωτογραφική μηχανή, σε εικόνα σε κλίμακα του γκρι. Ο βασικός λόγος για αυτή τη μετατροπή, είναι ότι με την μετατροπή αυτή μειώνεται αισθητά το πλήθος των υπολογισμών που πρέπει να εκτελέσει ο αλγόριθμος και το μέγεθος της μνήμης που πρέπει να δεσμεύσει για την λειτουργία του. Όπως προαναφέρθηκε στο Κεφάλαιο 1.2, το κάθε color space αποτελείται από τρία ή τέσσερα χρωματικά συστατικά. Αυτό σημαίνει πως σε κάθε εικόνα αντιστοιχούν τόσοι δυσδιάστατοι πίνακες, όσα και τα συστατικά του color space.

Οι εικόνες στην κλίμακα του γκρι είναι συχνά το αποτέλεσμα της μέτρησης της έντασης του φωτός σε κάθε pixel σε μία ενιαία ζώνη του ηλεκτρομαγνητικού φάσματος (π.χ. υπέρυθρο, ορατό φως, υπεριώδες φως, κλπ.) και είναι μονοχρωματικά ορθές μόνο όταν συλλαμβάνεται μία συγκεκριμένη συχνότητα. Στην ουσία, η εικόνα στην κλίμακα του γκρι, είναι μια εικόνα στην οποία η αξία του κάθε pixel είναι ένα μόνο δείγμα που φέρει μόνο την πληροφορία της έντασης του φωτός.

Αυτό σημαίνει πως για κάθε μια εικόνα έχουμε έναν δυσδιάστατο πίνακα και με πρόχειρους υπολογισμούς μπορούμε να διαπιστώσουμε ότι η μετατροπή της RGB

εικόνας σε κλίμακα του γκρι μειώνει το πλήθος των υπολογισμών που πρέπει να εκτελέσει ο αλγόριθμος στο ένα τρίτο.

Η εντολή που χρησιμοποιήθηκε είναι η εξής:
`orim1=rgb2gray(orim);`

Βήμα 3

Συχνά η επιφάνεια του σκυροδέματος παρουσιάζει ατέλειες και ανωμαλίες. Στην προτεινόμενη μέθοδο, σχεδιάστηκε ένα μεσαίο φίλτρο, για να μειώσει την επίδραση των ανωμαλιών της επιφάνειας στον αλγόριθμο ανίχνευσης ρωγμών. Ένα φίλτρο 5*5 εφαρμόστηκε στην εικόνα. Το μεσαίο φίλτρο είναι μια μη γραμμική λειτουργία, η οποία χρησιμοποιείται στην επεξεργασία εικόνας για να μειώσει το θόρυβο και να εξομαλύνει την επιφάνεια. Το μεσαίο φίλτρο εφαρμόζεται ως μάσκα με λειτουργία συνέλιξης στην εικόνα. Η συνέλιξη είναι μια πράξη που συσχετίζει δύο συναρτήσεις, ή εν προκειμένω εικόνες. Ειδικότερα, το κύριο χαρακτηριστικό της είναι πως λαμβάνει υπόψη της όχι μόνο το τρέχον pixel, αλλά όλα τα γειτονικά pixel, ώστε να προκύψει ένα μέσο αποτέλεσμα. Στην πράξη, η συνέλιξη μοντελοποιεί τη διέλευση ενός σήματος από ένα φίλτρο. Έχοντας ένα φίλτρο 5*5, το κέντρο αυτού του πίνακα(του μεσαίου φίλτρου) μετακινείται σε κάθε στοιχείο της εικόνας. Με άλλα λόγια, το κέντρο του φίλτρου τοποθετείται σε κάθε στοιχείο της εικόνας ξεχωριστά, επικαλύπτοντας με 25 pixels. Η μέση τιμή των 25 pixels υπολογίζεται και αντικαθιστά την αρχική ένταση. Για τις εντάσεις των συνόρων, καθορίζονται εντάσεις με μηδενική τιμή, η οποία ονομάζεται μηδενική παρεμβολή.

Η εντολή που χρησιμοποιήθηκε είναι η εξής:
`grim=medfilt2(orim1,[5 5]);`

Βήμα 4

Τα pixels ρωγμών στην κλίμακα του γκρι δείχνουν μια ξαφνική αλλαγή της έντασής τους σε σχέση με τα γειτονικά τους. Ως εκ τούτου, η εύρεση αυτών των στοιχείων της εικόνας, θα δώσει έμφαση στις ρωγμές. Έτσι εφαρμόστηκε η μέθοδος ανίχνευσης ακμών Sobel στην εικόνα. Ο αλγόριθμος Sobel ανιχνεύει τη μεταβολή της έντασης της εικόνας στα γειτονικά pixel. Το φιλτράρισμα έγινε χρησιμοποιώντας τη λειτουργία συνέλιξης για κάθε κατεύθυνση (x και y κατεύθυνση) ξεχωριστά. Στη συνέχεια, οι δύο φιλτραρισμένες εικόνες συνδυάζονται σε κάθε θέση και το αποτέλεσμα είναι μια εικόνα με γκρίζο επίπεδο με ενισχυμένες άκρες.

Το φίλτρο Sobel μπορεί να σχεδιαστεί σύμφωνα με τη λειτουργία Sobel ως εξής:

$$G(x) = \begin{matrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{matrix}, \quad G(y) = \begin{matrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{matrix} \quad (1)$$

Για να ορίσουμε το φίλτρο sobel χρησιμοποιήθηκαν οι παρακάτω εντολές:
`Hx=fspecial('sobel');`
`Hx=Hx';`

Τα παράγωγα προς κάθε κύρια κατεύθυνση (x και y), ορίζονται από την εξ. (1). Εφαρμόζονται στην εικόνα της κλίμακας του γκρι χρησιμοποιώντας τη λειτουργία συνέλιξης με μηδενική επένδυση. Στη συνέχεια, ο παρακάτω τύπος χρησιμοποιήθηκε για να πάρουμε την εικόνα των ακμών :

$$I(x, y) = |I_x(x, y)| + |I_y(x, y)| \quad (2)$$

Όπου, $|I_x(x, y)|$ και $|I_y(x, y)|$ είναι φιλτραρισμένα αποτελέσματα G_x και G_y στην εικόνα της κλίμακας του γκρι, αντίστοιχα.

Η εντολή που χρησιμοποιήθηκε για να πραγματοποιήσει την παραπάνω διαδικασία είναι η εξής:

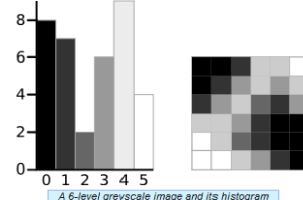
```
sobim=abs(imfilter(grim,Hx))+abs(imfilter(grim,Hy));
```

Βήμα 5

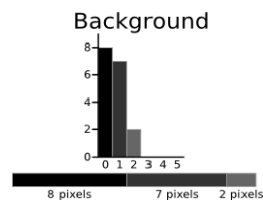
Το επόμενο βήμα είναι η μετατροπή της εικόνας από έγχρωμη σε ασπρόμαυρη. Σε προγραμματιστικό περιβάλλον αυτό σημαίνει ότι θέλουμε να μετατρέψουμε τον πίνακα που περιέχει τις τιμές των pixel, από δεκαδικό (0 έως 255) σε δυαδικό (0 ή 1). Οι τιμές 0 και 1 προσδιορίζουν αν το pixel είναι μαύρο ή λευκό, αντίστοιχα. Με τον τρόπο αυτό γίνεται πιο εύκολη η διαδικασία της επεξεργασίας, καθώς πλέον χρειάζεται να ελέγχουμε δύο πιθανές τιμές για κάθε στοιχείο του πίνακα, σε σχέση με τις 256 διαφορετικές τιμές που θα μπορούσε να έχει ως έγχρωμη φωτογραφία.

Η εντολή με την οποία γίνεται η παραπάνω διαδικασία είναι η `grethresh`. Η εντολή `graythresh` χρησιμοποιεί τη μέθοδο Otsu για να προσδιορίσει τη μέση τιμή της εικόνας, έτσι ώστε να μετατραπεί σε 0 και 1.

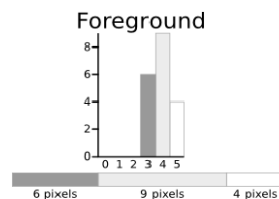
The algorithm will be demonstrated using the simple 6x6 image shown below. The histogram for the image is shown next to it. To simplify the explanation, only 6 greyscale levels are used.



The calculations for finding the foreground and background variances (the measure of spread) for a single threshold are now shown. In this case the threshold value is 3.



$$\begin{aligned} \text{Weight } W_b &= \frac{8+7+2}{36} = 0.4722 \\ \text{Mean } \mu_b &= \frac{(0 \times 8) + (1 \times 7) + (2 \times 2)}{17} = 0.6471 \\ \text{Variance } \sigma_b^2 &= \frac{((0 - 0.6471)^2 \times 8) + ((1 - 0.6471)^2 \times 7) + ((2 - 0.6471)^2 \times 2)}{17} \\ &= \frac{(0.4187 \times 8) + (0.1246 \times 7) + (1.8304 \times 2)}{17} \\ &= 0.4637 \end{aligned}$$



$$\begin{aligned} \text{Weight } W_f &= \frac{6+9+4}{36} = 0.5278 \\ \text{Mean } \mu_f &= \frac{(3 \times 6) + (4 \times 9) + (5 \times 4)}{19} = 3.8947 \\ \text{Variance } \sigma_f^2 &= \frac{((3 - 3.8947)^2 \times 6) + ((4 - 3.8947)^2 \times 9) + ((5 - 3.8947)^2 \times 4)}{19} \\ &= \frac{(4.8033 \times 6) + (0.0997 \times 9) + (4.8864 \times 4)}{19} \\ &= 0.5152 \end{aligned}$$

The next step is to calculate the 'Within-Class Variance'. This is simply the sum of the two variances multiplied by their associated weights.

$$\begin{aligned} \text{Within Class Variance } \sigma_W^2 &= W_b \sigma_b^2 + W_f \sigma_f^2 = 0.4722 * 0.4637 + 0.5278 * 0.5152 \\ &= 0.4909 \end{aligned}$$

Εικόνα 6. Μέθοδος Otsu

Σε μεγάλο αριθμό περιπτώσεων τα αντικείμενα που περιέχονται σε μια εικόνα με πολλά επίπεδα του γκρι, έχουν κοντινή μεταξύ τους φωτεινότητα. Αν προσδιοριστεί με μια κατάλληλη μέθοδο μια τιμή κατωφλιού, τότε θα μπορούμε να διαχωρίσουμε την εικόνα και να τη μετατρέψουμε σε δυαδική.

Η μέθοδος Otsu είναι ίσως η καλύτερη τεχνική εύρεσης κατωφλιού αν θέλουμε να διαχωρίσουμε μόνο δύο περιοχές φωτεινότητας. Όπως φαίνεται στην (Εικόνα 6), η μέθοδος Otsu χρησιμοποιεί έναν μαθηματικό αλγόριθμο ο οποίος βρίσκει τον αριθμό των pixels για τα διαφορετικά επίπεδα φωτεινότητας. Ελέγχει το εύρος τιμών και εν συνέχεια βρίσκει δύο μέσους όρους, έναν για κάθε εύρος τιμών. Τέλος η επιθυμητή τιμή που μας δίνει είναι ο μέσος όρος των δύο αυτών τιμών. Η τιμή αυτή βρίσκεται πάντα μεταξύ του 0 και του 1. Τα περιεχόμενα με μικρότερη τιμή από αυτή του κατωφλιού παίρνουν την τιμή 0, ενώ αν έχουν τιμή μεγαλύτερη από το επιθυμητό κατώφλι παίρνουν την τιμή 1.

Η εντολή που μετατρέπει την εικόνα από έγχρωμη σε ασπρόμαυρη είναι η εξής:
`binim1=im2bw(sobim,graythresh(sobim));`

Βήμα 6

Η επεξεργασμένη εικόνα που έχουμε σε αυτό το σημείο, συνήθως έχει μικρά αντικείμενα, τα οποία εμφανίζονται λόγω επιφανειακών ανωμαλιών. Για την κατάργηση αυτών των αντικειμένων, εφαρμόστηκε ένα όριο περιοχής στη δυαδική εικόνα. Ο αριθμός των pixels κάθε συνδεδεμένου στοιχείου (αντικείμενο) υπολογίζεται, και εκτιμάται η έκταση κάθε αντικειμένου. Τα αντικείμενα με έκταση περιοχής μικρότερη από 1000 pixels, θα προστεθούν στα pixel φόντου, δηλαδή θα πάρουν την τιμή 0 (μαύρα pixel).

Είναι δύσκολο να επιλεγεί ο αριθμός της έκτασης της περιοχής, που από αυτόν τον αριθμό και κάτω τα pixel θα προστίθενται στα pixel φόντου, και από αυτόν και πάνω θα κρατάνε την ήδη υπάρχουσα τιμή. Ο λόγος είναι ότι εξαρτάται από το μέγεθος της εικόνας και από τις ανωμαλίες που έχει η κάθε εικόνα. Από τον έλεγχο που πραγματοποιήθηκε στις επεξεργασμένες φωτογραφίες κρίθηκε ικανοποιητική η τιμή των 1000 pixel.

Η εντολές που πραγματοποιούν την παραπάνω διαδικασία είναι οι εξής:

```
L1=bwlabel(binim1);  
stat1=regionprops(binim1,'Area');  
area=cat(1,stat1.Area);  
v1=find(area>1000);  
binim2=ismember(L1,v1);
```

Βήμα 7

Ένα από τα σημαντικότερα προβλήματα στην αυτόματη ανίχνευση ρωγμών, είναι η ταξινόμηση μεταξύ ρωγμών και άλλων αντικειμένων στο σκυρόδεμα. Χαρακτηριστικά αντικείμενα είναι οι γραμμές φόντου, οι κηλίδες χρώματος, η υγρασία στο σκυρόδεμα, ο οπλισμός που σε ορισμένα σημεία μπορεί να φαίνεται λόγω κάποιου ελαττώματος, κλπ. Έτσι, στον αλγόριθμο προτείνονται και εφαρμόζονται πολλαπλές τεχνικές για να διαχωριστούν οι ρωγμές από άλλα αντικείμενα.

Οι ρωγμές δεν εμφανίζονται ως ευθείες γραμμές στην επιφάνεια του σκυροδέματος. Αντιθέτως, τα άκρα ενός σκυροδέματος, όπως μια στήλη ή μια δοκός, είναι συνήθως ευθείες.

Σε αυτό το βήμα, υπολογίζεται ο προσανατολισμός των συνδεδεμένων στοιχείων (αντικειμένων) στην εικόνα. Για κάθε αντικείμενο ξεχωριστά, αντιστοιχίζεται μια ισοδύναμη έλλειψη. Η γωνία μεταξύ του άξονα χ και του κύριου άξονα της ελλείψεως (μεγαλύτερης διαμέτρου), που κυμαίνεται από -90 έως 90 μοίρες, αναφέρεται ως ο προσανατολισμός κάθε αντικειμένου. Τα αντικείμενα με προσανατολισμό 0 , -90 και 90 μοίρες αναγνωρίζονται και προστίθενται στο παρασκήνιο. Μια μετατόπιση ± 1 βαθμού, θεωρείται δεδομένο ότι είναι απίθανο να έχει απολύτως κατακόρυφα και οριζόντια αντικείμενα.

Οι εντολές που υπολογίζουν τον προσανατολισμό κάθε αντικειμένου, και τοποθετούν τα αντικείμενα αυτά στο παρασκήνιο είναι η εξής:

```
stat2=regionprops(binim2,'Orientation');  
orientation=cat(1,stat2.Orientation);  
v2=find(abs(orientation)<89 & abs(orientation)>1)  
binim3=ismember(L2,v2);
```

Βήμα 8

Σε αυτό το βήμα, χρησιμοποιήθηκε μορφολογική λειτουργία, ούτως ώστε, κάποια pixel να αλλάξουν τιμή με βάση τα γειτονικά τους. Τα αντικείμενα (ρωγμές) θα μπορούσαν να αναγνωριστούν και σε προηγούμενα βήματα. Όμως παρατηρήθηκε ότι ενώ εκτελείται μορφολογική λειτουργία, κάποια άσχετα pixel μπορεί να συνδεθούν με τα pixel ρωγμών, γεγονός που μειώνει την ικανότητα του προγράμματος να ανιχνεύσει σωστά τις ρωγμές.

Για να αντιμετωπιστεί αυτό το ζήτημα, εφαρμόστηκε η μορφολογική λειτουργία “majority”. Ορίζεται μια δομή $3*3$ για να περάσει από την εικόνα. Εάν πέντε ή περισσότερα pixel στη δομή $3*3$ έχουν τιμή 1 (pixel ρωγμής), τότε όλη η δομή παίρνει την τιμή 1. Διαφορετικά, το πρόγραμμα ορίζει την τιμή 0 (pixel φόντου), και έτσι όλη η δομή παίρνει την τιμή 0, και τα pixel αυτά προστίθενται στο παρασκήνιο. Η διαδικασία αυτή επαναλαμβάνεται έως ότου να μην μπορεί να αλλάξει τιμή κανένα άλλο pixel.

Η εντολή με την οποία γίνεται αυτή η μορφολογική λειτουργία είναι η εξής:

```
binim4=bwmorph(binim3,'majority',1000)
```

Η εντολή είναι προγραμματισμένη να κάνει άπειρες επαναλήψεις, έως ότου να μην μπορεί να αλλάξει κάποιο άλλο pixel στην εικόνα. Στην εντολή υπάρχει όριο στις πόσες επαναλήψεις να τελειώνει (1000), διότι παρατηρήθηκε ότι από έναν αριθμό επαναλήψεων και πάνω δε άλλαζε κάτι σημαντικό στο αποτέλεσμα, παρά μόνο ότι καθυστερούσε το πρόγραμμα.

Βήμα 9

Η επεξεργασμένη εικόνα που υπάρχει σε αυτό το σημείο, συνήθως έχει κάποια μικρά αντικείμενα, που προέρχονται από την μορφολογική λειτουργία του προηγούμενου βήματος. Ανιχνεύονται αντικείμενα με αριθμό pixel μικρότερο από 200, παίρνουν την τιμή 0 (μαύρα pixel) και περνούν στο παρασκήνιο.

Η εντολή για την παραπάνω διαδικασία είναι η εξής:
`binim5=bwareaopen(binim4,200);`

Βήμα 10

Σε αυτό το βήμα, ανιχνεύονται και εξαλείφονται οι απόλυτες ευθείες. Παρόμοια λειτουργία έγινε και στο βήμα 7, με την διαφορά ότι στο βήμα 7 ανιχνευόταν ο προσανατολισμός των περιοχών, ενώ σε αυτό το βήμα ανιχνεύονται και εξαλείφονται οι απόλυτες ευθείες. Απόλυτες ευθείες μπορεί να είναι τα άκρα του σκυροδέματος, μια στήλη, μια δοκός ή ακόμα και ο σπλισμός της υποδομής που μπορεί να είναι εμφανείς σε κάποια σημεία.

Η μέθοδος που ακολουθήθηκε σε αυτό το βήμα, είναι ο μετασχηματισμός Hough. Ο μετασχηματισμός Hough είναι μια τεχνική που μπορεί να χρησιμοποιηθεί για να απομονώσει κάποια χαρακτηριστικά ενός συγκεκριμένου σχήματος σε μια εικόνα. Ειδικότερα, έχει χρησιμοποιηθεί για να εξάγει ευθείες, κύκλους και ελλείψεις (κωνικές τομές). Η απλούστερη μορφή μετασχηματισμού Hough είναι ο μετασχηματισμός γραμμής.

Για να χρησιμοποιηθεί ο μετασχηματισμός Hough, χρειάζεται έναν τρόπο χαρακτηρισμού μιας γραμμής. Ένας τρόπος αναπαράστασης της γραμμής είναι με την κλίση της γραμμής, δηλαδή:

$$y = m * x + b \quad (3)$$

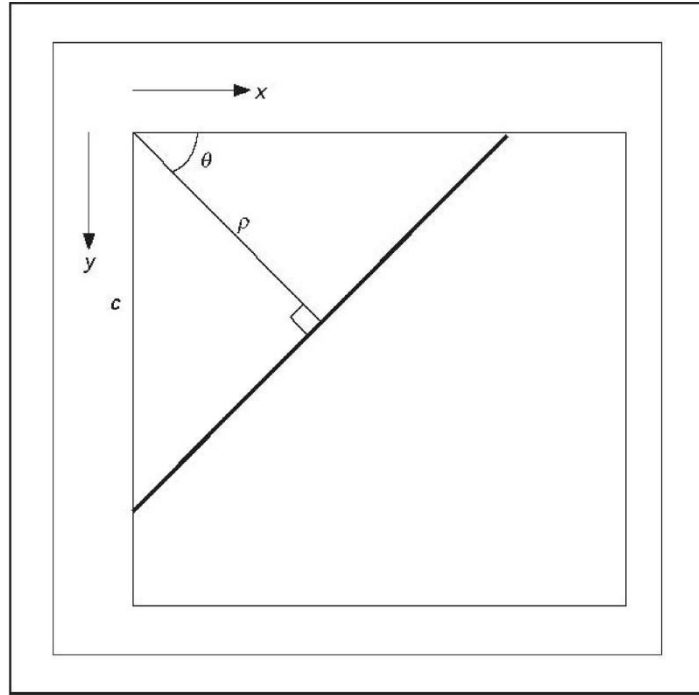
Όπου m είναι η κλίση της γραμμής και b είναι η y συντεταγμένη του σημείου της τομής της ευθείας με τον άξονα των y. Έχοντας αυτήν τη μορφή εξίσωσης της γραμμής, μπορούμε να εναλλαχθούμε μεταξύ οσωνδήποτε γραμμών που περνούν από ένα δοσμένο σημείο (x,y). Με εναλλαγή ανάμεσα σε συγκεκριμένες τιμές για το m, μπορούμε να λύσουμε ως προς b:

$$b = m * x + b \quad (4)$$

Παρ' όλα αυτά αυτή η μέθοδος δεν είναι πολύ σταθερή. Καθώς οι γραμμές τείνουν να γίνουν κάθετες, τα μεγέθη m και b τείνουν προς το άπειρο. Μια πιο χρήσιμη αναπαράσταση μιας γραμμής είναι μέσω της κανονικής της μορφής:

$$\rho = x * \cos \theta + y * \sin \theta \quad (5)$$

Η εξίσωση αυτή δηλώνει μια γραμμή που περνάει από το σημείο (x,y) και είναι κάθετη στη γραμμή που ξεκινάει από την αρχή των αξόνων και καταλήγει στο σημείο (ρ, θ) των πολικών συντεταγμένων (δηλαδή (ρ * cos θ , ρ * sin θ) στις καρτεσιανές συντεταγμένες), όπως φαίνεται στην παρακάτω εικόνα :



Εικόνα 7. Αναπαράσταση γραμμής σε κανονική μορφή

Για κάθε σημείο πάνω σε μία γραμμή, τα θ και ρ είναι σταθερά. Για κάθε δεδομένο σημείο (x,y) , παίρνουμε τις γραμμές που περνούν από αυτό λύνοντας ως προς ρ και θ . Εναλλάσσοντας ανάμεσα σε πιθανές τιμές της γωνίας θ , μπορούμε να λύσουμε με χρήση της εξίσωσης (5) απ' ευθείας ως προς ρ . Η μέθοδος αυτή αποδεικνύεται περισσότερο αποτελεσματική απ' ό,τι η (3), αφού είναι αριθμητικά σταθερή για αντιστοίχιση με γραμμές οποιασδήποτε γωνίας.

Οι εντολές που χρησιμοποιήθηκαν για να πραγματοποιηθεί ο μετασχηματισμός Hough, να βρεθούν οι απόλυτες ευθείες και να περάσουν στο παρασκήνιο είναι οι εξής:

```
[H,theta,rho]= hough(binim6,'Theta',[-90 0 89]);
peaks=houghpeaks(H,200);
lines=houghlines(binim6,theta,rho,peaks);
for m=1:length(lines)
    xy=[lines(m).point1; lines(m).point2];
    w=10;
    n=min(min(xy));
    if n<=w
        w=n-1;
    end
    n=max(max(xy));
    if w+n>=max(size(binim6))
        w=max(size(binim6))-n-1;
    end
    if abs(xy(1,2)-xy(2,2))/xy(1,2)<0.1
        binim6(xy(1,2)-w:xy(1,2)+w,xy(1,1):xy(2,1))=0;
    end
    if abs(xy(1,1)-xy(2,1))/xy(1,1)<0.1
        binim6(xy(1,2):xy(2,2), xy(1,1)-w:xy(1,1)+w)=0;
    end
end
end
```

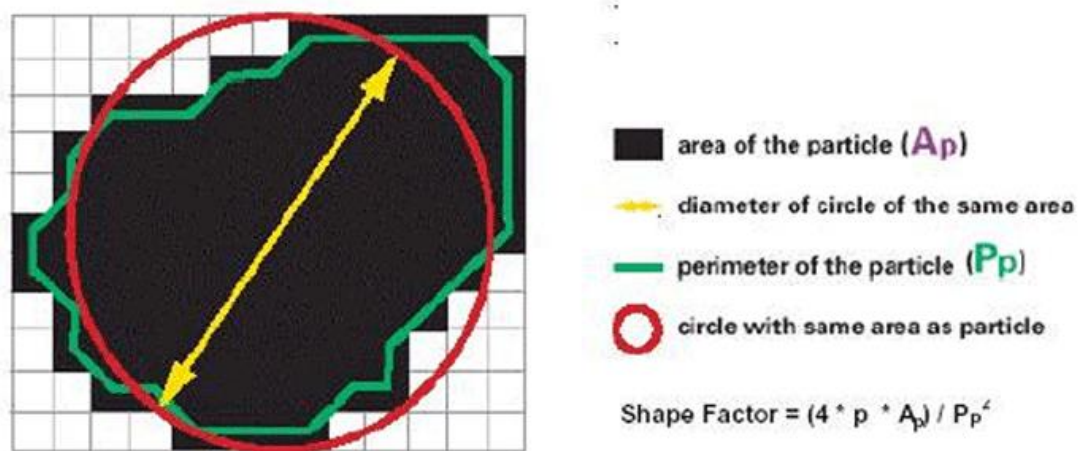

Αρχικά βρίσκουμε τα ρ και θ των ευθειών και στη συνέχεια βρίσκουμε τις ευθείες και τις περνάμε στο παρασκήνιο.

Βήμα 11

Σε αυτό το βήμα, χρησιμοποιήθηκε ο συντελεστής κυκλικότητας ο οποίος ορίζεται από :

$$SF = \frac{4 \cdot \pi \cdot A_p}{P_p^2} \quad (6)$$

Όπου, SF είναι ο συντελεστής κυκλικότητας, A_p είναι το μέγεθος της περιοχής και P_p είναι η περίμετρος της περιοχής.



Εικόνα 8. Συντελεστής κυκλικότητας

$$A_p = \pi \cdot r^2 \quad (7)$$

$$P_p = 2 \cdot \pi \cdot r \leftrightarrow r = \frac{P_p}{2 \cdot \pi} \quad (8)$$

$$A_p = \pi \cdot r^2 \leftrightarrow A_p = \pi \cdot \frac{P_p^2}{4 \cdot \pi^2} \leftrightarrow \frac{4 \cdot \pi \cdot A_p}{P_p^2} = 1 \quad (9)$$

Από την εξίσωση (9) βγαίνει πάντα αριθμός μεταξύ 0 και 1. Όταν το σχήμα είναι κύκλος το αποτέλεσμα είναι 1, ενώ όσο πιο κοντά είναι στο 0 το αποτέλεσμα, το σχήμα μοιάζει πιο πολύ με ευθεία.

Σε αυτό το βήμα χρησιμοποιήθηκε αυτός ο τύπος για να ελεγχθεί η κάθε περιοχή ξεχωριστά, και να αξιολογηθεί με τι μοιάζει η κάθε περιοχή. Έτσι, όταν μια περιοχή μοιάζει με κύκλο, δηλαδή το αποτέλεσμα της εξίσωσης (9) είναι κοντά στο 1, αλλάζει η τιμή της περιοχής (παίρνει την τιμή 0, μαύρα pixel), ενώ αν μια περιοχή μοιάζει με ευθεία, δηλαδή το αποτέλεσμα της εξίσωσης (9) είναι κοντά στο 0, δεν αλλάζει η τιμή της περιοχής, παραμένει ίδια ως πιθανή ρωγμή.

Η ομάδα εντολών που χρησιμοποιήθηκε στο συγκεκριμένο βήμα είναι η εξής:

```
stat3=regionprops(binim5, 'Area', 'Perimeter', 'PixelIdxList');
area1=[stat3.Area];
```

```

per=[stat3.Perimeter];
sfc=4*pi*area1./per.^2;
for k=1:length(sfc)
    if sfc(k)>0.1
        binim6(stat3(k).PixelIdxList)=0;
    end
end
end

```

Αυτό το βήμα είναι ουσιαστικά και η τελική επεξεργασία των εικόνων. Ότι παραμένει σε αυτό το στάδιο της επεξεργασίας των εικόνων είναι ρωγμή.

Βήμα 12

Τελευταίο βήμα είναι η αποθήκευση των εικόνων. Οι εντολές για την πραγματοποίηση αυτού του βήματος είναι οι εξής:

```

str1=['im_crack' num2str(i) '.jpg'];
imwrite(binim6,str1);

```

Μετά την ολοκλήρωση όλων των βημάτων του κώδικα ότι παραμένει στην τελική εικόνα θα είναι ρωγμή. Αυτό συμπεραίνεται από το γεγονός ότι έχουν εξαλειφτεί όλα τα άλλα ελαττώματα που μπορεί να έχει η εικόνα, όπως ευθείες γραμμές, μπογιά, υγρασία κ.α. Με τις μεθόδους που έχουν χρησιμοποιηθεί παραπάνω εξαλείφονται τα ελαττώματα, δηλαδή με τον μετασχηματισμό Hough οι γραμμές, με τον συντελεστή κυκλικότητας οι περιοχές που δεν μοιάζουν με ευθείες και οποιεσδήποτε άλλες ατέλειες.

Κεφάλαιο 3

3 Πειραματικά αποτελέσματα

Σε αυτό το κεφάλαιο θα συζητηθούν τα πειραματικά αποτελέσματα της προτεινόμενης μεθόδου, καθώς και τα πλεονεκτήματα και μειονεκτήματα της μεθόδου. Δοκιμάστηκε ο αλγόριθμος σε δείγμα 20 φωτογραφιών από τα κτήρια του ΤΕΙ Σίνδου. Η λήψη των φωτογραφιών έγινε με μια συμβατική ψηφιακή φωτογραφική μηχανή με ανάλυση 20-megapixels και στη συνέχεια επεξεργάστηκαν από τον αλγόριθμο. Επιλέχθηκαν διάφορες εικόνες προς επεξεργασία. Εικόνες από τοίχους, πεζοδρόμιο και άσφαλτο. Ο βασικός λόγος είναι ότι οι ρωγμές εμφανίζονται σε οποιαδήποτε κατασκευή σκυροδέματος ή ασφάλτου, και ελέγχθηκε αν το πρόγραμμα είναι σε θέση να αναγνωρίσει ρωγμές σε οποιαδήποτε κατασκευή από σκυρόδεμα.

Για το κάθε σημείο εξέτασης έγινε λήψη από διαφορετικές αποστάσεις (1m , 0.75m , 0.5m , 0.25m και 0.2m) της φωτογραφικής μηχανής από την υποδομή. Καθώς και διαφορετικό πάχος ρωγμών, για να ελεγχθεί αν ο αλγόριθμος είναι σε θέση να αναγνωρίσει μεγάλες αλλά και μικρές ρωγμές. Οι διαφορετικές αποστάσεις λήψης των φωτογραφιών γίνονται για να αποφασιστεί πιά είναι η καταλληλότερη απόσταση της φωτογραφικής μηχανής από την επιφάνεια της κατασκευής, ώστε το πρόγραμμα να ανιχνεύει με επιτυχία τις ρωγμές. Τα αποτελέσματα για την εύρεση ή μη των ρωγμών φαίνονται παρακάτω.

3.1 Πειραματικά αποτελέσματα της προτεινόμενης μεθόδου

Η πρώτη εικόνα που επεξεργάστηκε, είναι από το κτήριο του Τμήματος Αυτοματισμού. Τραβήχτηκαν 4 φωτογραφίες που αφορούν το ίδιο σημείο, σε τέσσερις διαφορετικές αποστάσεις της φωτογραφικής μηχανής από τον τοίχο. Οι αποστάσεις αυτές είναι 1m, 0.75m, 0.5m και 0.25m . Το πάχος της ρωγμής είναι περίπου 0.03mm και ο φωτισμός τη στιγμή της λήψης της φωτογραφίας είναι περίπου 1600 lux.

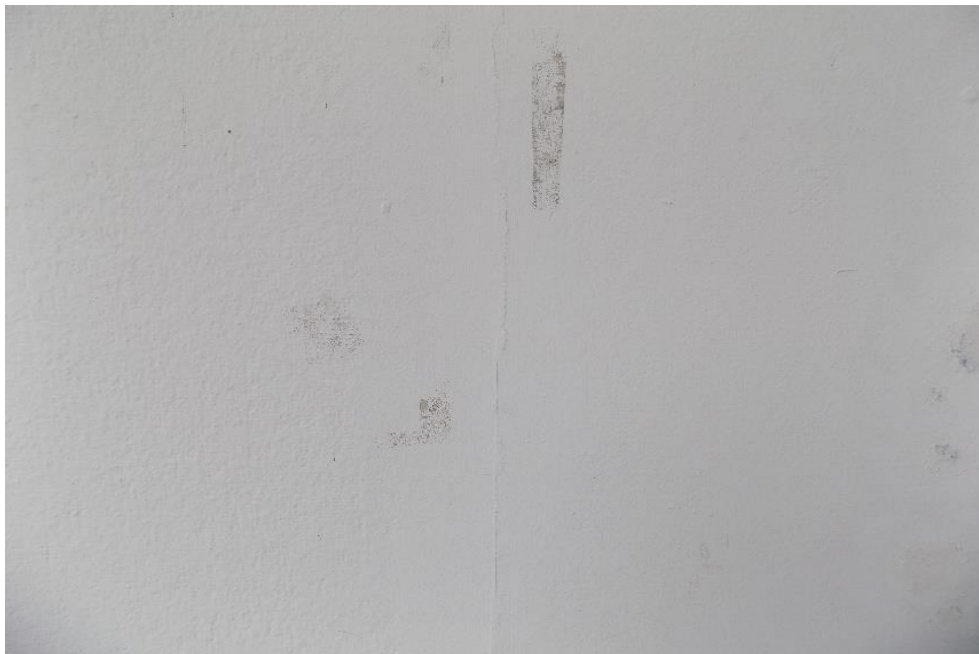
Στις παρακάτω εικόνες (Εικόνα 9, 10, 11 και 12), φαίνονται οι φωτογραφίες του τοίχου που τραβήχτηκαν από τις διαφορετικές αποστάσεις (1m, 0.75, 0.5, 0.25m) αντίστοιχα.



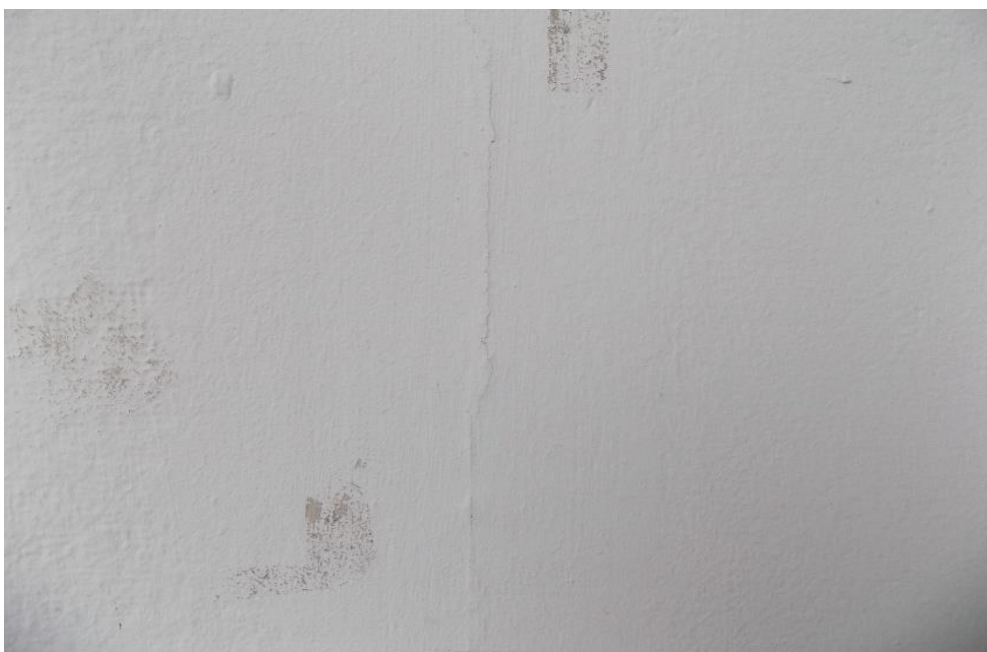
Εικόνα 9. Τοίχος Κτηρίου Αυτοματισμού 1m απόσταση



Εικόνα 10. Τοίχος Κτηρίου Αυτοματισμού 0.75m απόσταση

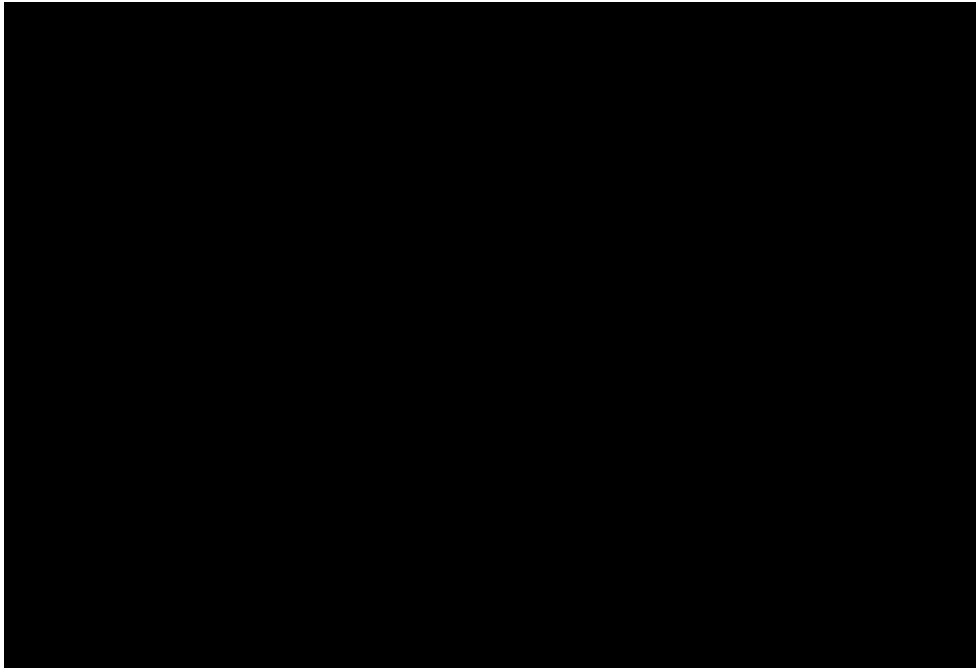


Εικόνα 11. Τοίχος Κτηρίου Αυτοματισμού 0.5m απόσταση

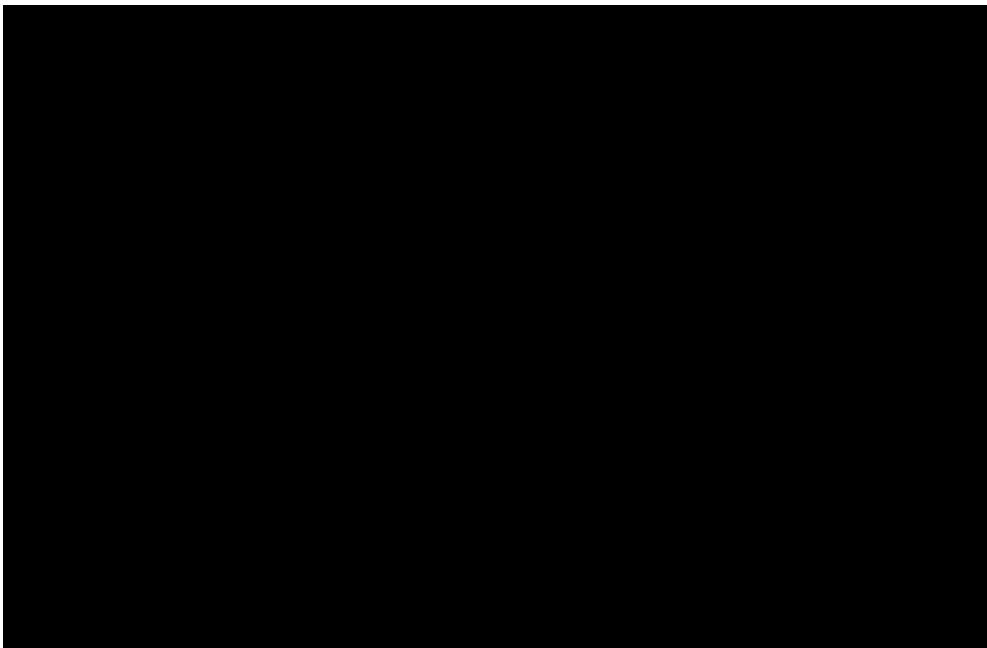


Εικόνα 12. Τοίχος Κτηρίου Αυτοματισμού 0.25m απόσταση

Παρακάτω παρατίθενται οι εικόνες (Εικόνα 13, 14, 15 και 16) οι οποίες είναι οι επεξεργασμένες εικόνες, των εικόνων (Εικόνα 9, 10, 11 και 12) αντίστοιχα.



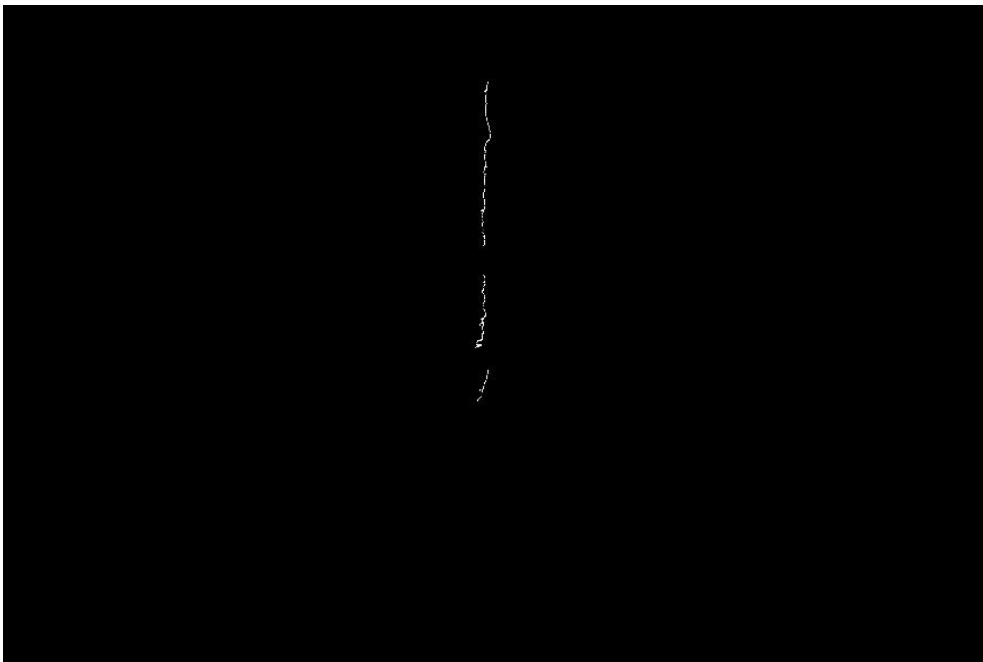
Εικόνα 13. Επεξεργασμένη εικόνα 9



Εικόνα 14. Επεξεργασμένη εικόνα 10



Εικόνα 15. Επεξεργασμένη εικόνα 11



Εικόνα 16. Επεξεργασμένη εικόνα 12

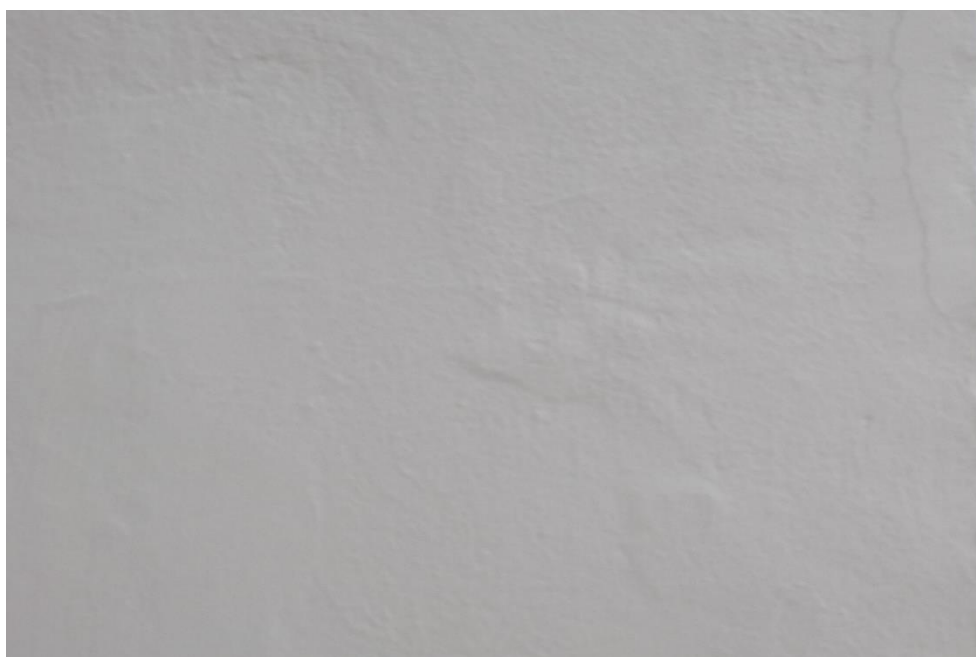
Όπως φαίνεται στις αρχικές εικόνες, υπάρχει μια κάθετη ρωγμή στη μέση περίπου της εικόνας. Η ρωγμή έχει πάχος περίπου 0.03mm (πολύ μικρή). Στις Εικόνες

9 και 10 παρατηρήθηκε ότι η ρωγμή δεν φαίνεται σχεδόν καθόλου, διότι είναι πολύ μικρή και η απόσταση της φωτογραφικής μηχανής είναι αρκετά μακριά ώστε να είναι ευδιάκριτη η ρωγμή. Κάτι πού φαίνεται και στις επεξεργασμένες εικόνες (Εικόνα 13 και 14), το πρόγραμμα δεν είναι σε θέση να αναγνωρίσει την ρωγμή. Αντίθετα στις εικόνες 11 και 12, η ρωγμή φαίνεται πιο καθαρά, επειδή η απόσταση της φωτογραφικής μηχανής είναι πιο κοντά (0.5m και 0.25m αντίστοιχα). Το ίδιο παρατηρήθηκε και στις επεξεργασμένες εικόνες (Εικόνα 15 και 16), όπου το πρόγραμμα βγάζει αποτέλεσμα και φαίνεται σχεδόν όλη η ρωγμή.

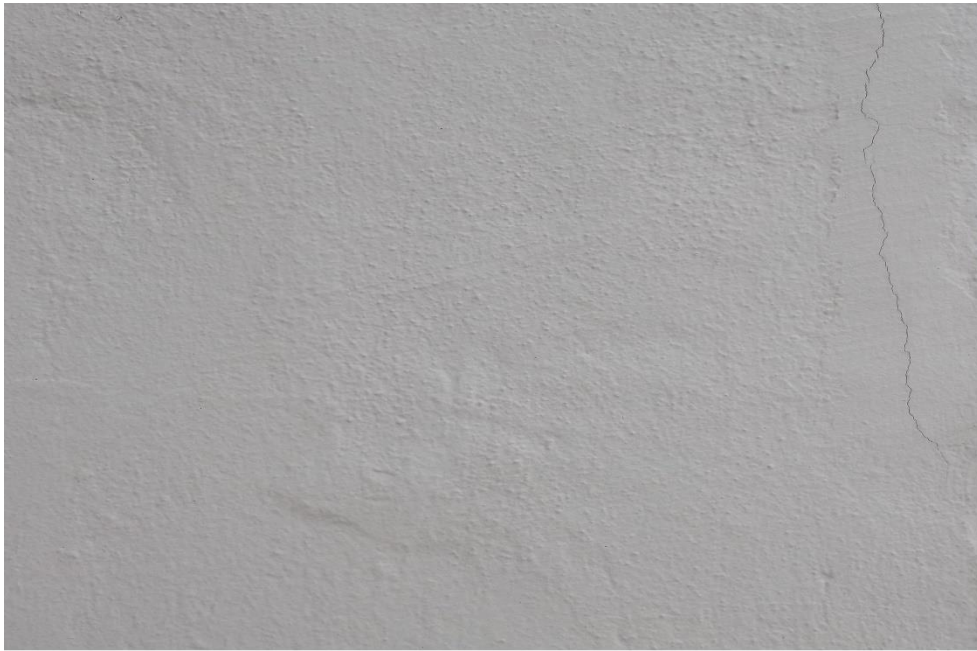
Στις αρχικές εικόνες φαίνονται κάποιες κηλίδες χρώματος και βρωμιάς. Το πρόγραμμα είναι σε θέση να εξαλείψει αυτές τις ατέλειες διότι αυτές οι περιοχές δεν μοιάζουν με ρωγμή.

Οι επόμενες φωτογραφίες που επεξεργάστηκαν είναι από τοίχο του κτηρίου της Διοίκησης. Τραβήχτηκαν τέσσερις φωτογραφίες του ίδιου σημείου από διαφορετικές αποστάσεις 1m, 0.75m, 0.5m και 0.25m. Η ρωγμή έχει πάχος περίπου 0.1mm και ο φωτισμός την στιγμή της λήψης των φωτογραφιών ήταν 2600 lux.

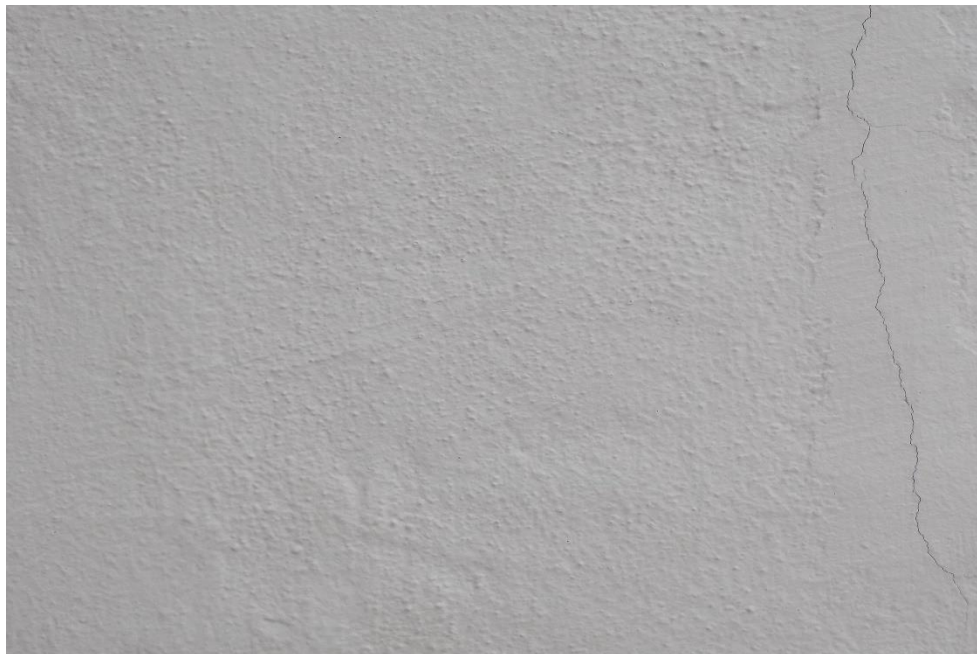
Στις παρακάτω εικόνες φαίνονται οι τέσσερις φωτογραφίες από τις τέσσερις διαφορετικές αποστάσεις από τον τοίχο.



Εικόνα 17. Τοίχος Κτηρίου Διοίκησης 1m απόσταση



Εικόνα 18. Τοίχος Κτηρίου Διοίκησης 0.75m απόσταση

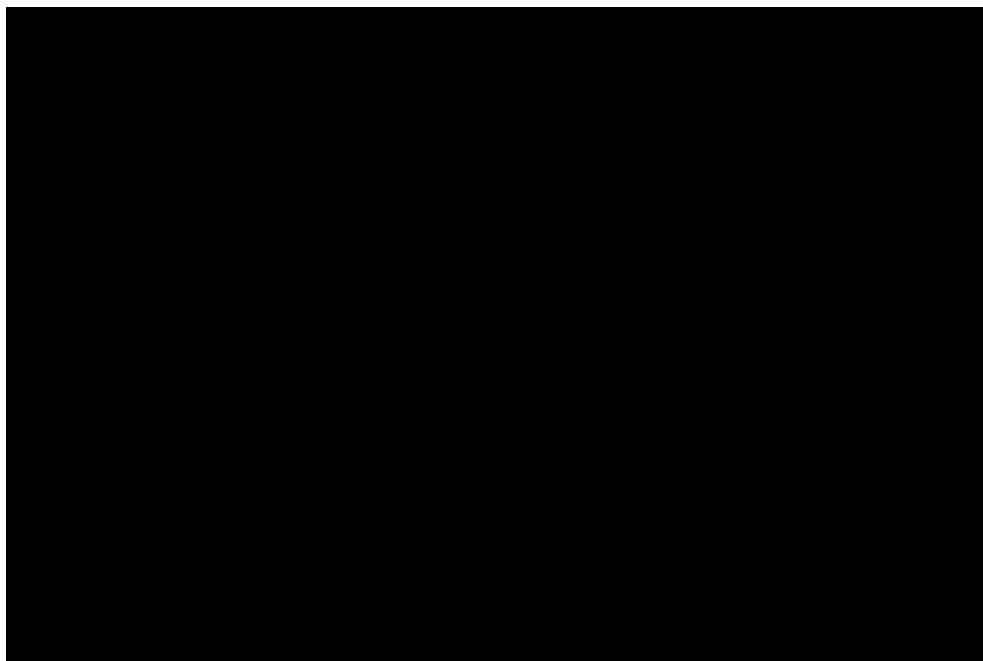


Εικόνα 19. Τοίχος Κτηρίου Διοίκησης 0.5m απόσταση

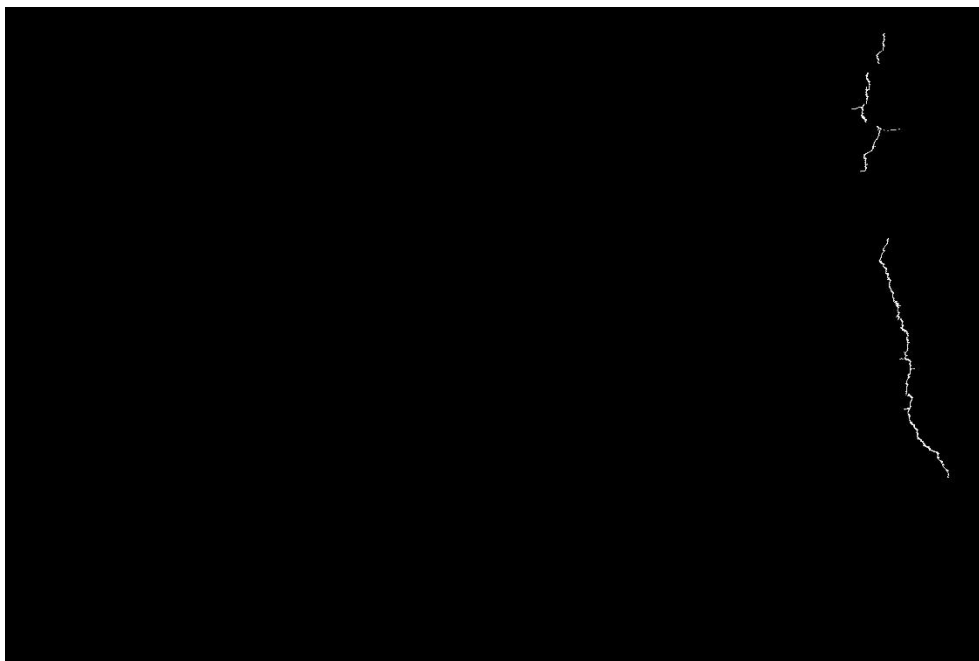


Εικόνα 20. Τοίχος Κτηρίου Διοίκησης 0.25m απόσταση

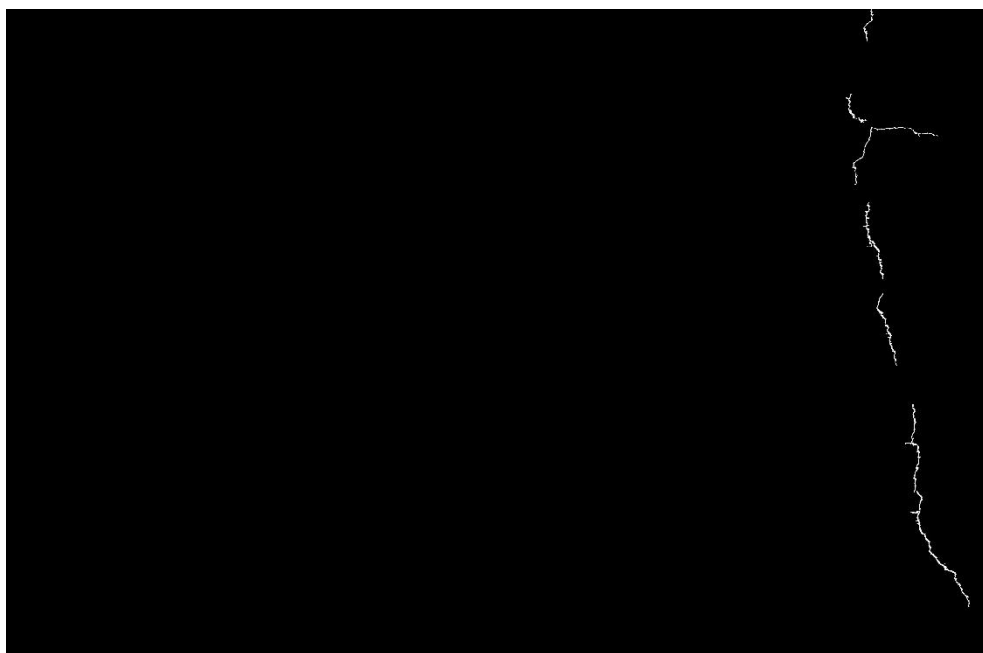
Οι παρακάτω εικόνες (Εικόνα 21, 22, 23 και 24) είναι οι επεξεργασμένες εικόνες των αρχικών (Εικόνα 17, 18, 19 και 20) αντίστοιχα.



Εικόνα 21. Επεξεργασμένη εικόνα 17



Εικόνα 22. Επεξεργασμένη εικόνα 18



Εικόνα 23. Επεξεργασμένη εικόνα 19



Εικόνα 24. Επεξεργασμένη εικόνα 20

Στις αρχικές εικόνες φαίνεται μια ρωγμή δεξιά της εικόνας. Παρατηρήθηκε ότι στην πρώτη εικόνα (Εικόνα 17) η ρωγμή δεν είναι ευδιάκριτη, κάτι που φαίνεται και στην επεξεργασμένη εικόνα (Εικόνα 21), στην οποία δεν έχει ανιχνευτεί καμία ρωγμή. Αυτό οφείλεται στην απόσταση της φωτογραφικής μηχανής από τον τοίχο. Αντίθετα στις επόμενες εικόνες (Εικόνα 18, 19 και 20) η ρωγμή είναι ευδιάκριτη, κάτι που φαίνεται και στις επεξεργασμένες εικόνες αυτών (Εικόνα 22, 23 και 24).

Στην Εικόνα 22 παρατηρήθηκε ότι η ρωγμή που έχει ανιχνευτεί δεν είναι ολόκληρη και υπάρχουν σημεία ασυνέχειας. Δηλαδή υπάρχει ρωγμή σε αυτά τα σημεία αλλά το πρόγραμμα δεν την έχει ανιχνεύσει, κάτι που οφείλεται στην απόσταση της φωτογραφικής μηχανής και στο ότι η ρωγμή δεν έχει σε όλη της την έκταση το ίδιο πάχος. Στις δύο επόμενες εικόνες (Εικόνα 23 και 24), όπου η ρωγμή είναι ξεκάθαρη και η λήψη της φωτογραφίας έχει γίνει από πολύ κοντινή απόσταση (0.5m και 0.25m), έχει ανιχνευτεί πλήρως η ρωγμή.

Οι επόμενες φωτογραφίες που επεξεργάστηκαν είναι από τον τοίχο του κτηρίου Οχημάτων. Έγινε λήψη πέντε φωτογραφιών σε διαφορετικές αποστάσεις (1m, 0.75m, 0.5m, 0.25m και 0.2m). Το πάχος της ρωγμής είναι περίπου 0.2mm και ο φωτισμός τη στιγμή της λήψης ήταν 90lux.

Στις παρακάτω εικόνες (Εικόνα 25, 26, 27, 28 και 29) παρατηρήθηκε ότι η ρωγμή, ειδικά στις δύο πρώτες φωτογραφίες (Εικόνα 25 και 26), είναι δυσδιάκριτη. Αυτό οφείλετε στην απόσταση της φωτογραφικής μηχανής από τον τοίχο αλλά και το

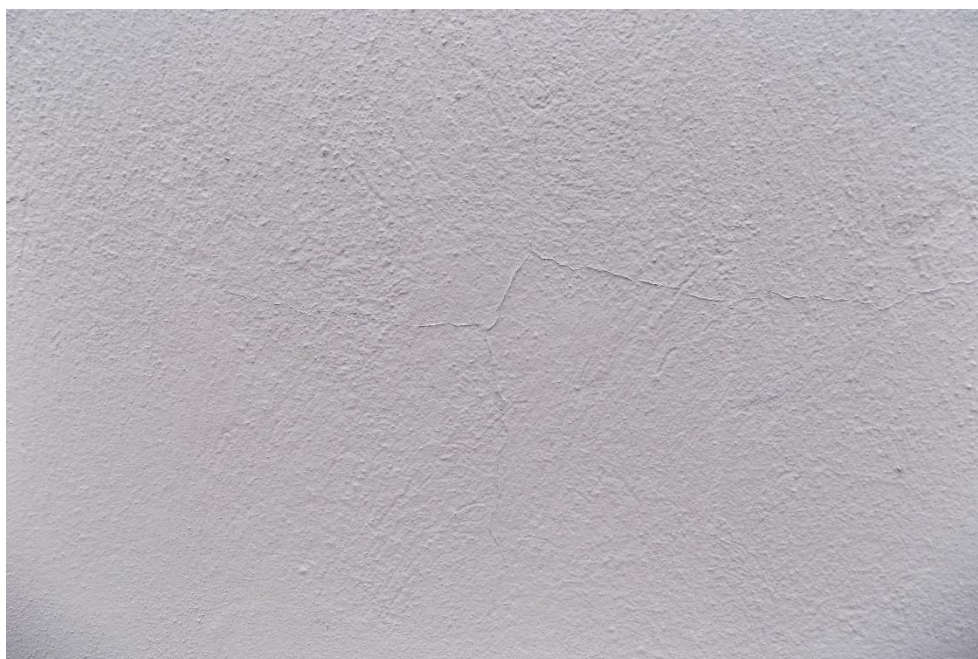
γεγονός ότι η συγκεκριμένη ρωγμή είναι επιφανειακή και πρέπει να ελεγχθεί από πολύ κοντινή απόσταση για να διαγνωστεί σωστά εάν υπάρχει ελάττωμα.



Εικόνα 25. Τοίχος Κτηρίου Οχημάτων 1m απόσταση



Εικόνα 26 Τοίχος Κτηρίου Οχημάτων 0.75m απόσταση



Εικόνα 27. Τοίχος Κτηρίου Οχημάτων 0.5m απόσταση

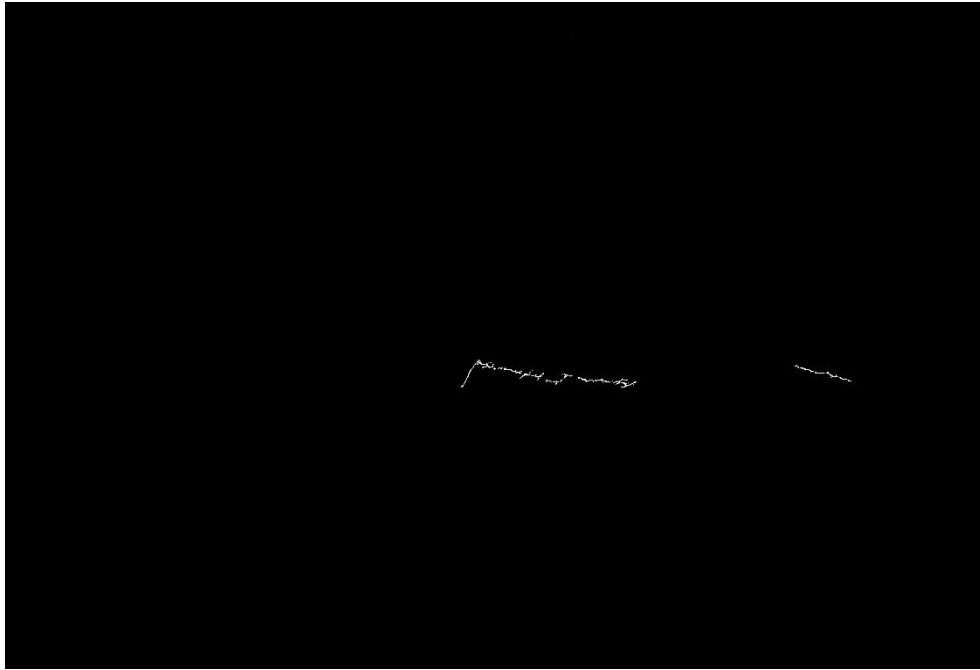


Εικόνα 28. Τοίχος Κτηρίου Οχημάτων 0.25m απόσταση



Εικόνα 29. Τοίχος Κτηρίου Οχημάτων 0.2m απόσταση

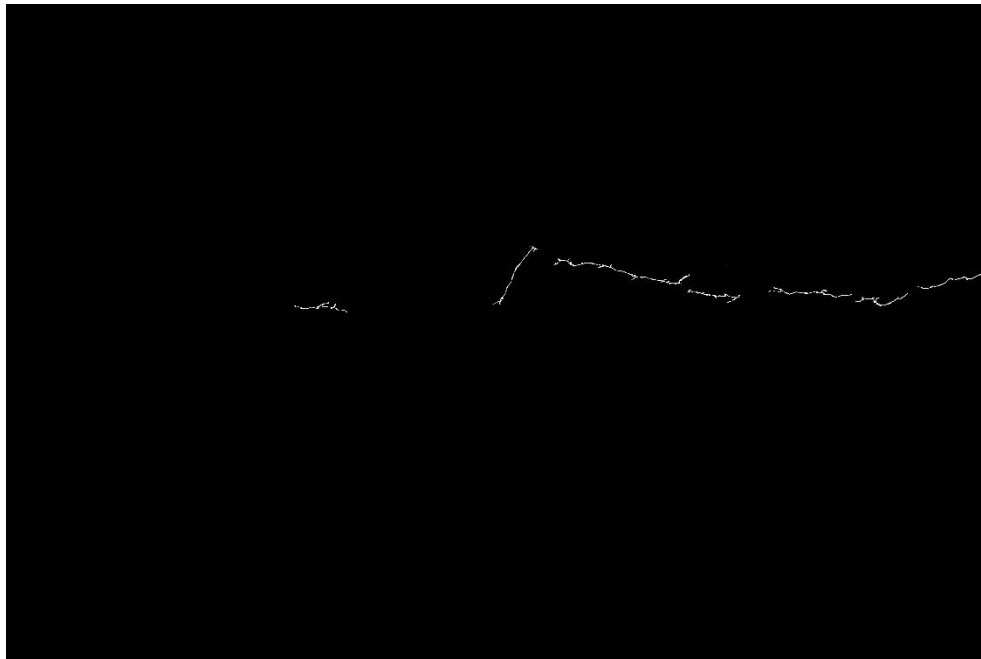
Παρακάτω φαίνονται οι επεξεργασμένες εικόνες των αρχικών φωτογραφιών (Εικόνα 30, 31, 32, 33 και 34) αντίστοιχα.



Εικόνα 30. Επεξεργασμένη εικόνα 25



Εικόνα 31. Επεξεργασμένη εικόνα 26



Εικόνα 32. Επεξεργασμένη εικόνα 27



Εικόνα 33. Επεξεργασμένη εικόνα 28



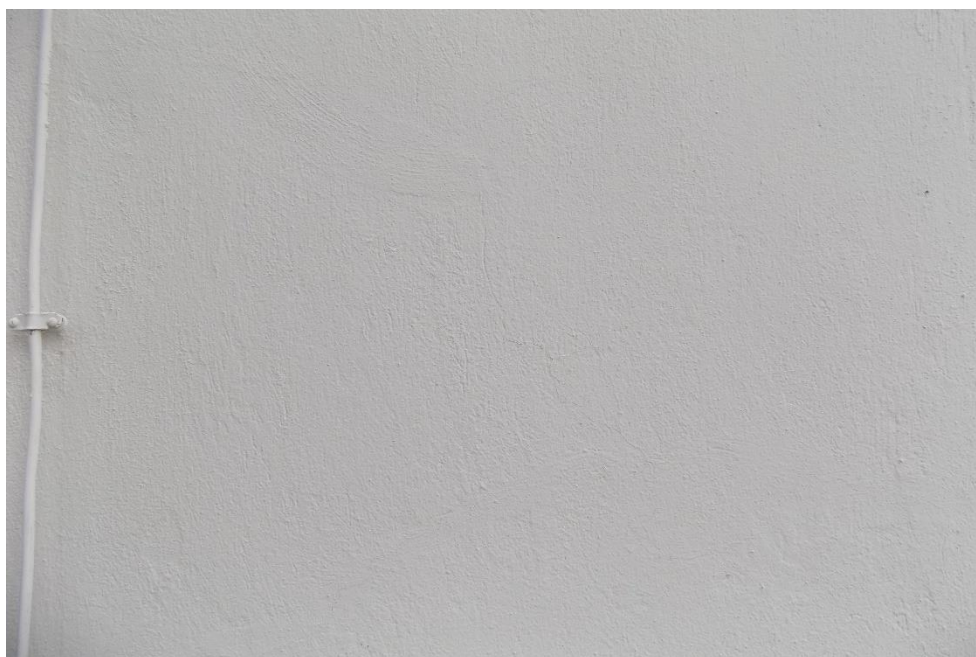
Εικόνα 34. Επεξεργασμένη εικόνα 29

Όπως προαναφέρθηκε, στις δύο πρώτες εικόνες (Εικόνα 25 και 26), η ρωγμή είναι δυσδιάκριτη λόγω του τύπου της ρωγμής αλλά και της απόστασης της φωτογραφικής μηχανής. Όμως το πρόγραμμα είναι σε θέση να διακρίνει, μερικώς, και αυτές τις ρωγμές κάτι που παρατηρήθηκε και στις Εικόνες 30 και 31 αντίστοιχα. Παρατηρήθηκε όμως μια ασυνέχεια στη ρωγμή, δηλαδή η ρωγμή δεν ανιχνεύεται πλήρως, κάτι που είναι λογικό να συμβεί καθώς η ρωγμή είναι δυσδιάκριτη ακόμα και από κάποιον έμπειρο επιθεωρητή. Η ρωγμή είναι ευδιάκριτη εάν κάνουμε ζουμ στην εικόνα.

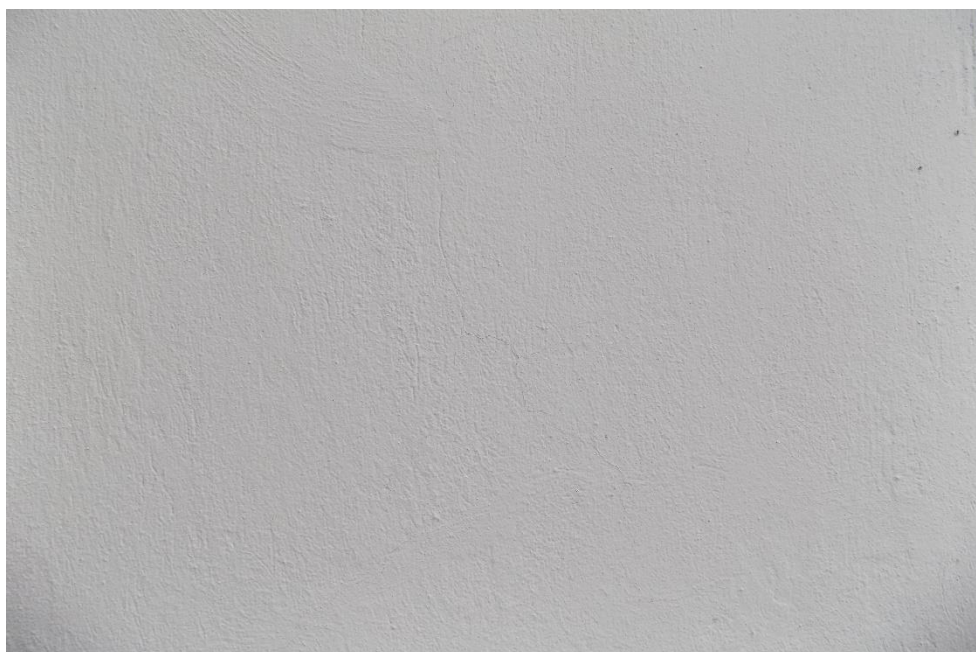
Αντίθετα στις άλλες τρεις φωτογραφίες (Εικόνα 27, 28 και 29), όπου η ρωγμή είναι ευδιάκριτη αφού η λήψη έχει γίνει από κοντινή απόσταση, το πρόγραμμα είναι σε θέση να ανιχνεύσει σωστά την ρωγμή (Εικόνα 32, 33 και 34). Βέβαια, και σε αυτές τις εικόνες υπάρχει μια μικρή ασυνέχεια στις ρωγμές, κάτι που είναι λογικό να συμβεί καθώς ο τοίχος δεν είναι ομαλός και επίσης σε κάποια σημεία η ρωγμή έχει μικρότερο πάχος με αποτέλεσμα το πρόγραμμα να μην μπορεί να ανιχνεύσει αυτές τις πολύ λεπτές ρωγμές.

Στη συνέχεια έγινε λήψη πέντε φωτογραφιών από άλλο τοίχο του κτηρίου Οχημάτων, από πέντε διαφορετικές αποστάσεις (1m, 0.75m, 0.5m, 0.25m και 0.2m). Το πάχος της ρωγμής σε αυτόν τον τοίχο είναι περίπου 0.1mm και ο φωτισμός τη στιγμή της λήψης ήταν 360lux.

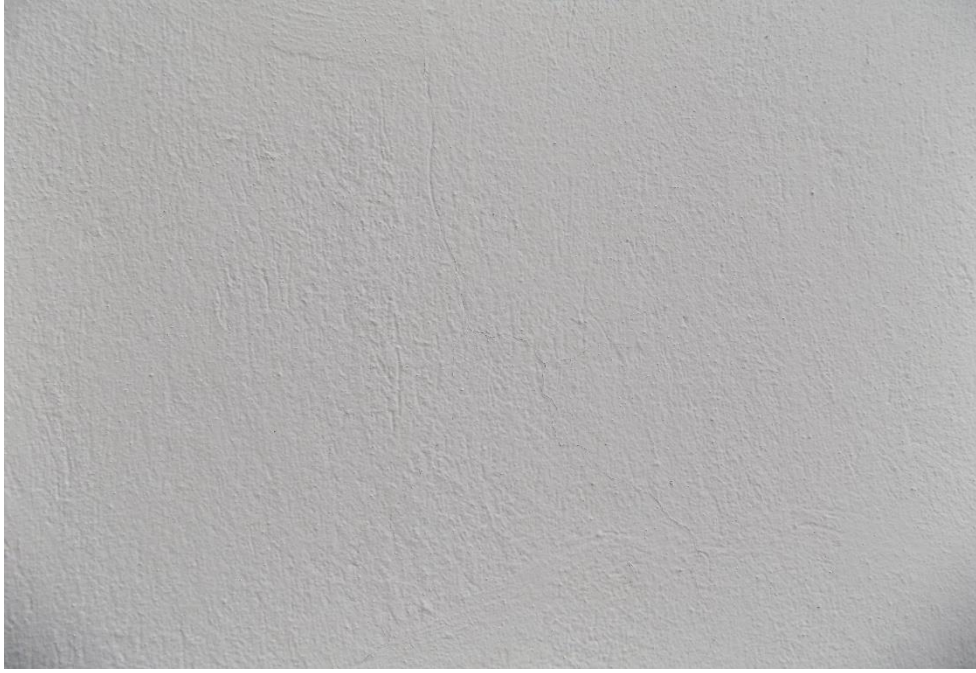
Παρακάτω φαίνονται οι πέντε φωτογραφίες (Εικόνα 35, 36, 37, 38 και 39), από το κτήριο Οχημάτων από τις πέντε διαφορετικές αποστάσεις (1m, 0.75m, 0.5m, 0.25m και 0.2m) αντίστοιχα.



Εικόνα 35. Τοίχος Κτηρίου Οχημάτων 1m απόσταση



Εικόνα 36. Τοίχος Κτηρίου Οχημάτων 0.75m απόσταση



Εικόνα 37. Τοίχος Κτηρίου Οχημάτων 0.5m απόσταση



Εικόνα 38. Τοίχος Κτηρίου Οχημάτων 0.25m απόσταση



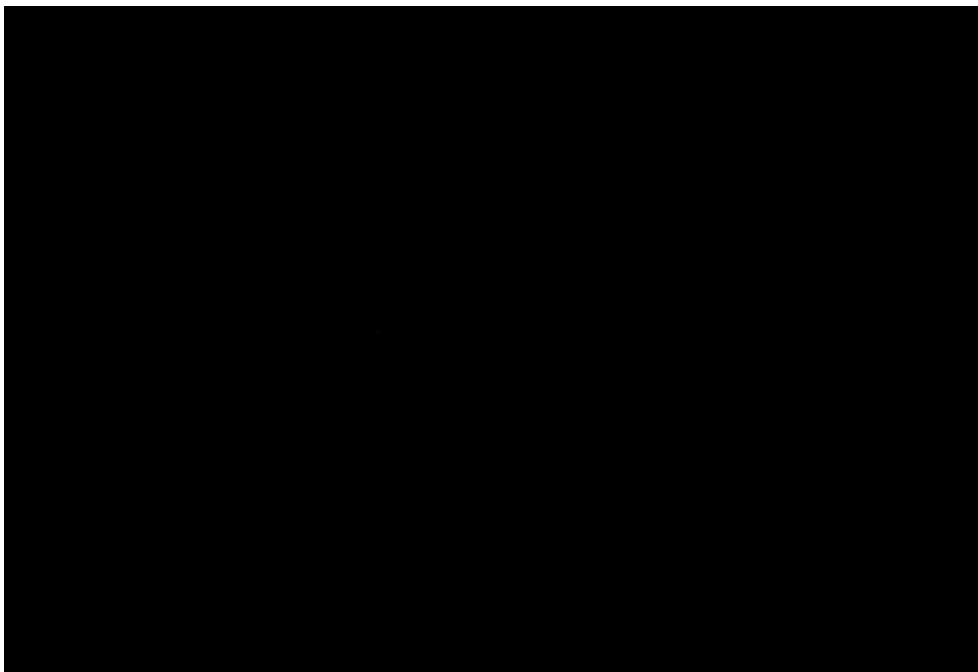
Εικόνα 39. Τοίχος Κτηρίου Οχημάτων 0.2m απόσταση

Στις τρεις πρώτες εικόνες (Εικόνα 35, 36 και 37), παρατηρήθηκε ότι υπάρχει ρωγμή αλλά είναι δυσδιάκριτη λόγω της απόστασης της λήψης, η ρωγμή διακρίνεται μόνο με ζουμ στις εικόνες αυτές. Αντίθετα στις άλλες δύο εικόνες (Εικόνα 38 και 39), η ρωγμή διακρίνεται σχετικά εύκολα.

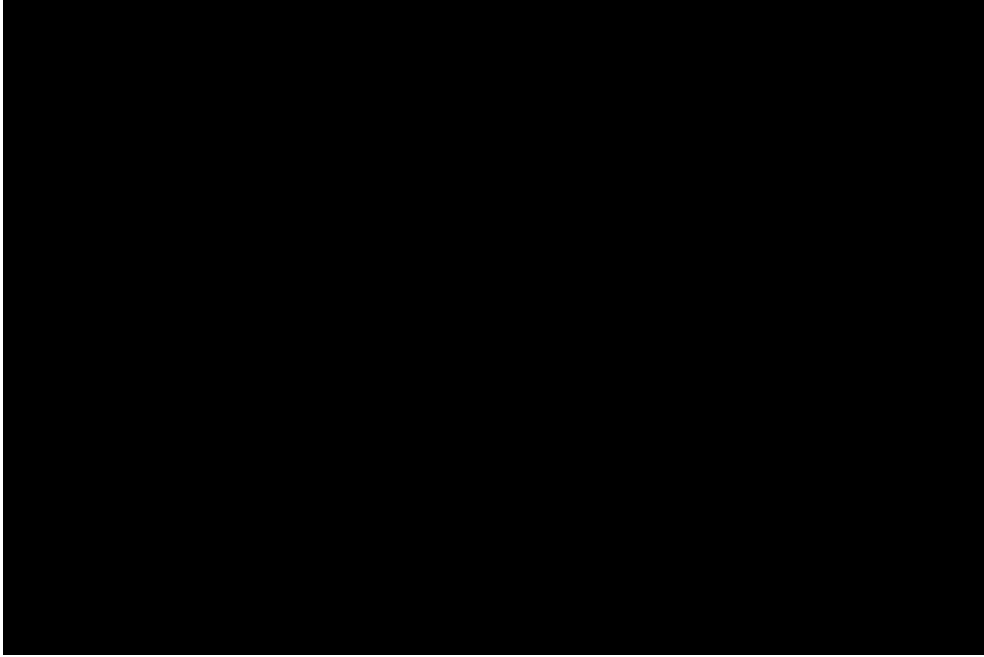
Παρακάτω φαίνονται οι επεξεργασμένες εικόνες (Εικόνα 40, 41, 42, 43 και 44), των αρχικών φωτογραφιών (Εικόνα 35, 36, 37, 38 και 39), και οι οποίες δίνουν το τελικό αποτέλεσμα για την ύπαρξη ή όχι ρωγμής.



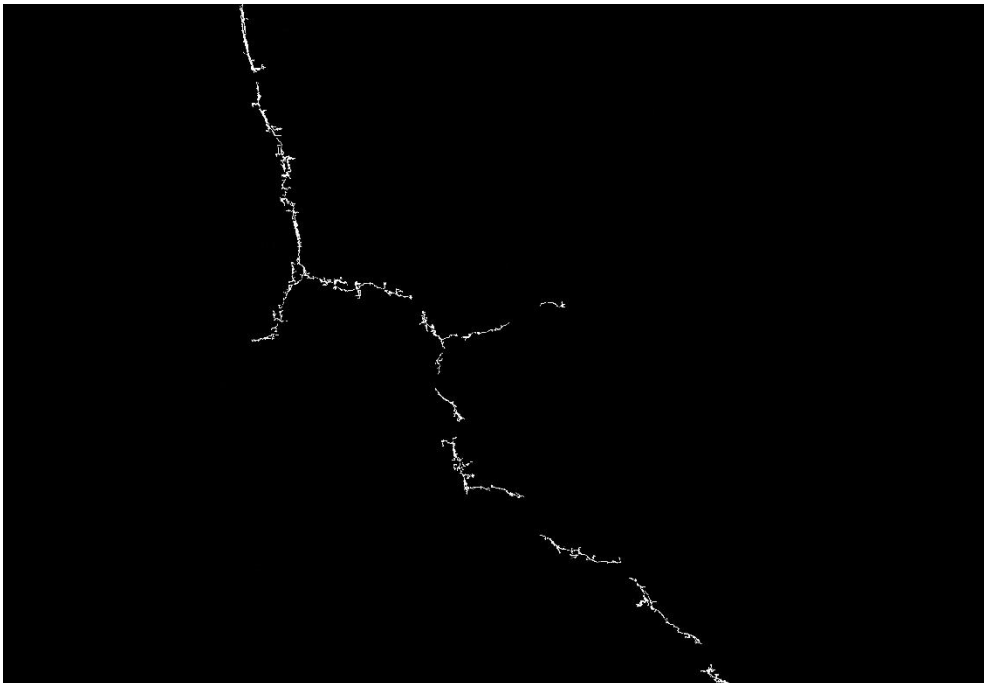
Εικόνα 40. Επεξεργασμένη εικόνα 35



Εικόνα 41. Επεξεργασμένη εικόνα 36



Εικόνα 42. Επεξεργασμένη εικόνα 37



Εικόνα 43. Επεξεργασμένη εικόνα 38



Εικόνα 44. Επεξεργασμένη εικόνα 39

Στις τρεις πρώτες επεξεργασμένες εικόνες (Εικόνα 40, 41 και 42), παρατηρήθηκε ότι το πρόγραμμα δεν έχει ανιχνεύσει κάποια ρωγμή, κάτι που είναι λογικό αφού οι ρωγμές στις πρώτες τρεις εικόνες (Εικόνα 35, 36 και 37) είναι δυσδιάκριτες. Βέβαια, το σημαντικό που παρατηρήθηκε στην πρώτη εικόνα (Εικόνα 35), είναι ότι στα αριστερά της εικόνας φαίνεται ένας σωλήνας ο οποίος κατά την επεξεργασία της εικόνας αφαιρέθηκε από το πρόγραμμα σαν ευθεία. Ο σωλήνας αφαιρέθηκε από το τελικό αποτέλεσμα, στο βήμα (10) του κώδικα σαν απόλυτη ευθεία.

Αντίθετα στις δύο τελευταίες εικόνες (Εικόνα 43 και 44), οι οποίες είναι από αρκετά κοντινή απόσταση, το πρόγραμμα ανίχνευσε σωστά την ρωγμή. Με κάποια ασυνέχεια βέβαια, κάτι που είναι λογικό όπως προαναφέρθηκε, λόγω της ανωμαλίας της επιφάνειας του τοίχου και ότι σε κάποια σημεία η ρωγμή ήταν πολύ λεπτή για να ανιχνευτεί από το πρόγραμμα.

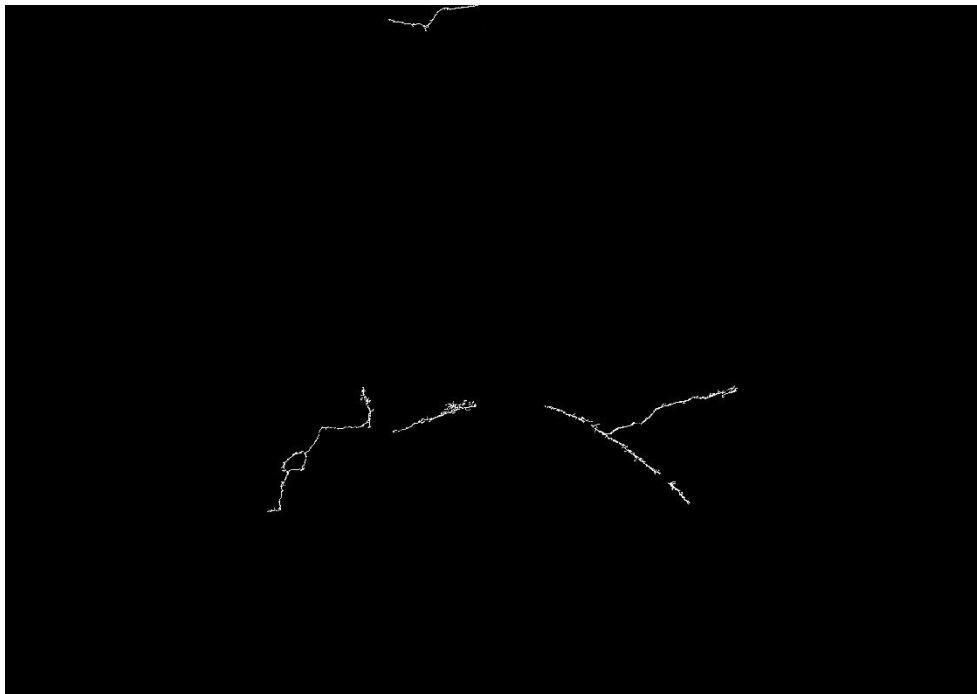
Επόμενη εικόνα που επεξεργάστηκε είναι από πεζοδρόμιο έξω από το κτήριο Πληροφορικής Αυτοματισμού. Η απόσταση της φωτογραφικής μηχανής από το πεζοδρόμιο είναι 1m. Η ρωγμή έχει πάχος περίπου 2mm και ο φωτισμός την στιγμή της λήψης της φωτογραφίας ήταν 10000lux.

Ο βασικός λόγος που επιλέχθηκε η επεξεργασία και φωτογραφίας από πλάκες πεζοδρομίου είναι ελέγχθηκε αν ο αλγόριθμος μπορεί να εξαλείψει από το τελικό αποτέλεσμα τις απόλυτες ευθείες με επιτυχία, δηλαδή τις ενώσεις των πλακών. Παρακάτω φαίνεται η εικόνα του πεζοδρομίου που τραβήξαμε έξω από το κτήριο Πληροφορικής Αυτοματισμού (Εικόνα 45).



Εικόνα 45. Πλάκες πεζοδρομίου 1m απόσταση

Παρακάτω φαίνεται η επεξεργασμένη εικόνα (Εικόνα 46) της αρχικής εικόνας (εικόνα 45) του πεζοδρομίου.



Εικόνα 46. Επεξεργασμένη εικόνα 45

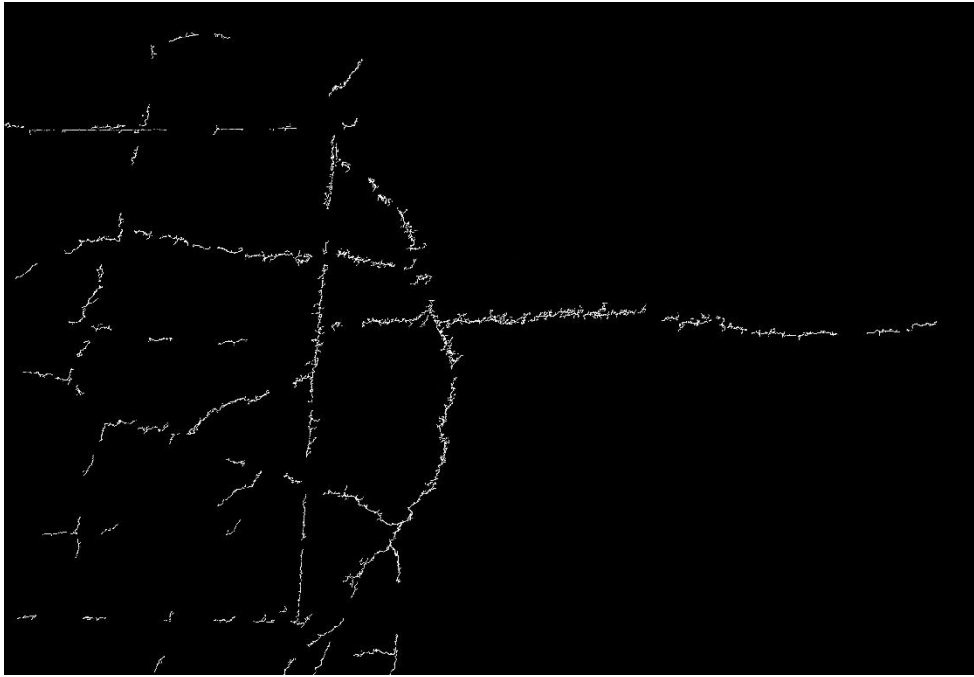
Όπως παρατηρήθηκε στην παραπάνω εικόνα (Εικόνα 46), το πρόγραμμα έχει ανιχνεύσει σωστά τις ρωγμές που υπάρχουν (σπάσιμο πλακών), και επίσης έχει εξαλείψει τις ενώσεις των πλακών από τα βήματα (7 και 10), που χωρίς αυτά θα ανιχνευόταν σαν ρωγμές. Επίσης παρατηρήθηκε ότι παρόλο που ο φωτισμός δεν είναι ίδιος σε όλη την εικόνα και ότι οι πλάκες δεν έχουν την ίδια απόχρωση μεταξύ τους, στο κάτω μέρος της εικόνας φαίνεται πως οι πλάκες έχουν υγρασία, το πρόγραμμα είναι σε θέση να ανιχνεύσει ρωγμές ακόμα και σε κακές εικόνες χωρίς κανένα πρόβλημα.

Η τελευταία εικόνα προς επεξεργασία (Εικόνα 47), είναι από δρόμο έξω από το κτήριο Πληροφορικής Αυτοματισμού. Επιλέχθηκε και μια εικόνα δρόμου, καθώς η ανίχνευση ελαττωμάτων στους δρόμους είναι πολύ σημαντικό στο να διορθώνονται τα προβλήματα έγκαιρα για να μην υπάρξει κάποιο ατύχημα λόγω προβλήματος στο οδόστρωμα. Η λήψη της εικόνας έγινε από απόσταση 1m, με πάχος ρωγμών 4-5mm και φωτισμό τη στιγμή της λήψης 10000lux.



Εικόνα 47. Ασφαλτος 1m απόσταση

Στην Εικόνα 47 παρατηρήθηκαν πολλές ρωγμές σε πολύ κοντινή απόσταση, ομάδα ρωγμών, που πιθανόν να προκλήθηκαν από καθίζηση του υποστρώματος κάτω από την ασφαλτο με αποτέλεσμα την δημιουργία ρωγμών.



Εικόνα 48. Επεξεργασμένη εικόνα 47

Στην παραπάνω εικόνα (Εικόνα 48), παρατηρήθηκε ότι έχουν ανιχνευτεί σχεδόν πλήρως οι ρωγμές. Βέβαια, στο τελικό αποτέλεσμα φαίνεται μια ασυνέχεια μεταξύ των ρωγμών, δηλαδή σε κάποια σημεία οι ρωγμές διακόπτονται, κάτι που είναι λογικό να συμβαίνει καθώς το πάχος των ρωγμών σε ορισμένα σημεία γίνεται πολύ μικρό και επίσης η επιφάνεια του οδοστρώματος δεν είναι καθόλου ομαλή. Η ανωμαλία του οδοστρώματος διορθώνεται μέχρις ένα βαθμό στο βήμα (3), όπου χρησιμοποιήθηκε ένα μεσαίο φίλτρο για να εξομαλυνθεί η επιφάνεια, αλλά δεν γίνεται πλήρως ομαλή ώστε το πρόγραμμα να είναι σε θέση να ανιχνεύσει απόλυτα τις ρωγμές του οδοστρώματος.

Κεφάλαιο 4

4 Συμπεράσματα και μελλοντική εργασία

Στην παρούσα εργασία προτείνεται μια αποτελεσματική μέθοδος για την αυτόματη ανίχνευση επιφανειακών ρωγμών σε δομές σκυροδέματος. Η προτεινόμενη μέθοδος βασίστηκε στο έργο των Sattar Dorafshan, Marc Maguire και Xiaojun Qi “Automatic Surface Crack Detection in Concrete Structures Using OTSU Thresholding and Morphological Operations” (2016)”. Κατηγοριοποιήθηκε ως μέθοδος χωρίς εκπαίδευση. Ο αλγόριθμος που χρησιμοποιήθηκε είναι αρκετά απλός και φιλικός προς τον χρήστη. Η προτεινόμενη μέθοδος είναι σε θέση να εντοπίσει άμεσα τις ρωγμές ενώ οι περισσότερες από τις τρέχουσες τεχνικές ανίχνευσης ρωγμών, χρησιμοποιούν βάσεις κατάρτισης και εκπαίδευσης για την λήψη αποφάσεων, κάτι που είναι χρονοβόρο και δαπανηρό.

4.1 Συμπεράσματα προτεινόμενης μεθόδου

Η προτεινόμενη μέθοδος επιτυγχάνει μεγάλο ποσοστό επιτυχίας στην ανίχνευση ρωγμών. Η επιτυχία της ανίχνευσης ρωγμών εξαρτάται από διάφορους παράγοντες, οι οποίοι είναι οι εξής:

- Πάχος ρωγμής
- Απόσταση της φωτογραφικής μηχανής από την επιφάνεια της υποδομής
- Η λήψη της φωτογραφίας να είναι καθαρή
- Ομαλότητα επιφάνειας
- Επαρκής φωτισμός τη στιγμή της λήψης της φωτογραφίας

Οι δύο πιο σημαντικοί παράγοντες στην ανίχνευση ρωγμών, από τους παραπάνω, είναι το πάχος της ρωγμής και η απόσταση της φωτογραφικής μηχανής από την επιφάνεια της υποδομής. Αυτοί οι δύο παράγοντες πρέπει να συνδυαστούν κατάλληλα ώστε η ρωγμή να φαίνεται καθαρά στις εικόνες.

Αν η απόσταση είναι μεγάλη και το πάχος της ρωγμής είναι πολύ μικρό (Εικόνα 9 και 10), η ρωγμή δεν φαίνεται καθαρά στην εικόνα και το πρόγραμμα δεν είναι σε θέση να αναγνωρίσει τις ρωγμές (Εικόνα 13 και 14). Πρέπει να συνδυάζονται σωστά και οι δύο αυτοί παράγοντες για ένα σωστό αποτέλεσμα. Όπως φαίνεται στις Εικόνες 30 και 31, το πρόγραμμα έχει ανιχνεύσει ένα πολύ μικρό κομμάτι της ρωγμής. Αυτό συμβαίνει γιατί η απόσταση της φωτογραφικής μηχανής από την επιφάνεια είναι αρκετά μακριά και έτσι δεν είναι ευδιάκριτη η ρωγμή (Εικόνα 25 και 26). Το ίδιο συμβαίνει και στις Εικόνες 35, 36 και 37, όπου η ρωγμή είναι 0.1mm (αρκετά μεγάλο πάχος) αλλά η απόσταση της φωτογραφικής μηχανής είναι αρκετά μακριά (1m, 0.75m και 0.5m) για το συγκεκριμένο πάχος ρωγμής ώστε να είναι ευδιάκριτη η ρωγμή.

Αρκετά σημαντικό είναι και η καθαρότητα της φωτογραφίας. Όπως βλέπουμε και στην Εικόνα 17, η εικόνα είναι αρκετά θολή και η ρωγμή είναι δυσδιάκριτη. Κάτι που βλέπουμε και στην επεξεργασμένη εικόνα (Εικόνα 21), το πρόγραμμα δεν έχει βγάλει κάποιο αποτέλεσμα, αρκετά λογικό καθώς η ρωγμή δεν είναι ευδιάκριτη.

Επίσης σημαντικός παράγοντας είναι και η ομαλότητα της επιφάνειας. Αν μια επιφάνεια είναι ομαλή, όπως ένας τοίχος συνήθως, το πρόγραμμα είναι πιο πιθανό να ανιχνεύσει μια ρωγμή σωστά χωρίς να ανιχνεύει θόρυβο λόγω ανωμαλίας της επιφάνειας. Αν όμως μια επιφάνεια έχει ανωμαλίες, όπως για παράδειγμα η άσφαλτος που η επιφάνειά της δεν είναι ομαλή (Εικόνα 47), το πρόγραμμα είναι αρκετά πιθανό να βρίσκει λάθη ή μη σωστή εύρεση όλων των ρωγμών. Όπως φαίνεται και στην Εικόνα 48, το πρόγραμμα λόγω των ανωμαλιών της ασφάλτου δεν έχει ανιχνεύσει πλήρως τις ρωγμές. Διακρίνονται κάποιες ασυνέχειες μεταξύ των ρωγμών. Βέβαια, το σημαντικό είναι ότι έχει ανιχνεύσει ένα μεγάλο ποσοστό των ρωγμών σωστά.

4.2 Μελλοντική εργασία

Σε αυτό το κεφάλαιο, θα αναλυθούν και θα προταθούν ενδεχόμενες βελτιώσεις του προγράμματος, ώστε να είναι σε θέση να αναγνωρίζει πλήρως τις ρωγμές. Η βασική βελτίωση που πρέπει να γίνει, είναι το πρόγραμμα να είναι σε θέση να αναγνωρίζει ρωγμές πολύ μικρές, ρωγμές σε ανώμαλες επιφάνειες καθώς και να μπορεί να εξαλείψει πλήρως τις ατέλειες που μπορεί να έχει η κάθε εικόνα, όπως σκιές, γραμμές φόντου, μπογιά, κ.α., που μπορεί να μπερδέψουν το πρόγραμμα.

Οι αυτοματοποιημένες μεθοδολογίες ανίχνευσης ρωγμών με βάση την εικόνα, είναι σχετικά νέες για την δομική μηχανική και έχουν τεράστιες δυνατότητες για μελλοντικές μελέτες.

Μια άλλη μέθοδος μπορεί να οριστεί με βάση τον αλγόριθμο εκπαίδευσης για την επίλυση ενός συγκεκριμένου προβλήματος. Κατασκευάζεται πρώτα μια βάση δεδομένων, η οποία χωρίζεται σε δύο υποσύνολα : το υποσύνολο με εικόνες με ρωγμές και το υποσύνολο με εικόνες χωρίς ρωγμές. Για κάθε υποσύνολο, ορίζεται ένας χαρακτηριστικός δείκτης, όπως ο παράγοντας σχήματος με την μορφή μιας γραμμής απόφασης. Η βάση δεδομένων κατάρτισης έχοντας αρκετές εικόνες σε ένα σύνολο δεδομένων κατάρτισης, το πρόγραμμα θα είναι σε θέση να συγκρίνει έναν νέο δείκτη χαρακτηριστικών εικόνων με τη γραμμή απόφασης και να τον αντιστοιχεί σε ένα από τα υποσύνολα. Τα συστήματα νευρωνικών δικτύων και ομαδοποίησης είναι από τις πιο δημοφιλείς τεχνικές κατάρτισης για να ορίσουν τη γραμμή αποφάσεων.

Το κατώτατο όριο της περιοχής κατωφλιού μπορεί να βελτιωθεί για να γίνει πιο δυναμικό με την χρήση των εντάσεων των pixels σε κάθε εικόνα αντί να έχει έναν στατικό αριθμό όπως έχει εφαρμοστεί σε αυτή την εργασία. Η συγκεκριμένη μέθοδος δεν είναι σε θέση να αναγνωρίσει μικρές ρωγμές που είναι σημαντικά μικρότερες από το πλάτος των pixels, ακόμα και αν είναι πολύ μεγαλύτερες από κάποια pixels. Μικρές ρωγμές που ταξινομούνται ως ρωγμές κόπωσης σε κατασκευές από σκυρόδεμα, είναι πολύ συχνές και μπορεί να προκύψουν από επαναλαμβανόμενη δυναμική φόρτωση και

εκφόρτωση. Οι ρωγμές κόπωσης παίζουν σημαντικό ρόλο στη διαρθρωτική παρακολούθηση της υγείας τόσο στις κατασκευές από χάλυβα, όσο και από σκυρόδεμα. Ωστόσο, οι τρέχοντες αυτοματοποιημένοι αλγόριθμοι δεν είναι αποτελεσματικοί στην αναγνώριση τους. Οι βελτιώσεις σε αυτούς τους τομείς θα οδηγήσουν σε πιο εκτεταμένη υιοθέτηση της αυτοματοποιημένης ανίχνευσης ρωγμών με βάση την επεξεργασία εικόνων στον τομέα της δομικής μηχανικής.

Παράρτημα

Κώδικας Matlab

```
Hx=fspecial('sobel');
% ορισμός φίλτρου sobel για την ανίχνευση οριζόντιων γραμμών
Hy=Hx';
% ορισμός φίλτρου sobel για την ανίχνευση κατακόρυφων γραμμών
for i=1:20
% έναρξη βρόγχου, κάθε i για κάθε αρχική εικόνα
    str=['image' num2str(i) '.jpg'];
    orim=imread(str);
    % διάβασμα εικόνων από τον φάκελο που είναι αποθηκευμένες
    orim1=rgb2gray(orim);
    % μετατροπή της RGB εικόνας σε κλίμακα του γκρι
    grim=medfilt2(orim1,[5 5]);
    % εφαρμόζεται ένα μεσαίο φίλτρο για την εξομάλυνση της επιφάνειας
    sobim=abs(imfilter(grim,Hx))+abs(imfilter(grim,Hy));
    % εφαρμόζεται ανιχνευτής Sobel για την ενδυνάμωση των άκρων της
    % εικόνας
    binim1=im2bw(sobim,graythresh(sobim));
    % εφαρμόζεται η μέθοδος κατωφλιού Otsu για την μετατροπή της
    % εικόνας σε ασπρόμαυρη, για την απόκτηση της δυαδικής εικόνας
    L1=bwlabel(binim1);
    stat1=regionprops(binim1,'Area');
    area=cat(1,stat1.Area);
    v1=find(area>1000);
    binim2=ismember(L1,v1);
    % αναγνώριση περιοχών με μέγεθος μικρότερο από 1000 pixels, και
    % τοποθέτηση αυτών στο παρασκήνιο
    L2=bwlabel(binim2);
    stat2=regionprops(binim2,'Orientation');
    orientation=cat(1,stat2.Orientation);
    v2=find(abs(orientation)<89 & abs(orientation)>1);
    binim3=ismember(L2,v2);
    % αναγνώριση περιοχών με προσανατολισμό 0, 90 και -90 μοίρες και
    % τοποθέτηση αυτών στο παρασκήνιο
    binim4=bwmorph(binim3,'majority',1000);
    % χρησιμοποιείται η μορφολογική λειτουργία "majority" για την
    % σύνδεση των κοντινών περιοχών μεταξύ τους
    binim5=bwareaopen(binim4,200);
    % αναγνώριση περιοχών με μέγεθος μικρότερο από 200 pixels, και
    % τοποθέτηση αυτών στο παρασκήνιο
    binim6=binim5;
    [H,theta,rho]=hough(binim6,'Theta',[-90 0 89]);
    peaks=houghpeaks(H,200);
    lines=houghlines(binim6,theta,rho,peaks);
    for m=1:length(lines)
        xy=[lines(m).point1; lines(m).point2];
        w=10;
        n=min(min(xy));
        if n<=w
            w=n-1;
        end
        n=max(max(xy));
        if w+n>=max(size(binim6))
            w=max(size(binim6))-n-1;
        end
        if abs(xy(1,2)-xy(2,2))/xy(1,2)<0.1
```

```

        binim6(xy(1,2)-w:xy(1,2)+w,xy(1,1):xy(2,1))=0;
    end
    if abs(xy(1,1)-xy(2,1))/xy(1,1)<0.1
        binim6(xy(1,2):xy(2,2), xy(1,1)-w:xy(1,1)+w)=0;
    end
end
% αναγνώριση περιοχών που είναι απόλυτες ευθείες, και τοποθέτηση
% αυτών στο παρασκήνιο
stat3=regionprops(binim6,'Area','Perimeter','PixelIdxList');
area1=[stat3.Area];
per=[stat3.Perimeter];
sfc=4*pi*area1./per.^2;
for k=1:length(sfc)
    if sfc(k)>0.1
        binim6(stat3(k).PixelIdxList)=0;
    end
end
% αναγνώριση περιοχών που μοιάζουν με κύκλο και τοποθέτηση αυτών
% στο παρασκήνιο, μέθοδος κυκλικότητας
str1=['im_crack' num2str(i) '.jpg'];
imwrite(binim6,str1);
% αποθήκευση επεξεργασμένων εικόνων σε φάκελο στον υπολογιστή
end
% τέλος βρόγχου

```


Βιβλιογραφία

1. Sattar Dorafshan, Marc Maguire, Xiaojun Qi, “Automatic Surface Crack Detection in Concrete Structures Using OTSU Thresholding and Morphological Operations”, UTC Report 01-2016, Utah State University, 2016
2. <https://www.mathworks.com>
3. Ευάγγελος Β. Χατζίκος, “Matlab για Επιστήμονες και Μηχανικούς”, Εκδόσεις Τζιόλα, 2013
4. [https://en.wikipedia.org/wiki/Shape_factor_\(image_analysis_and_microscopy\)](https://en.wikipedia.org/wiki/Shape_factor_(image_analysis_and_microscopy))
5. Yao Sun, “Automated Pavement Distress Detection Using Advanced Image Processing Techniques”, A Thesis, The University of Toledo, 2009