

ALGORITMI PRIMJENJIVI U POSTUPKU OČITAVANJA RADNIH AKTIVNOSTI S TAHOGRAFSKIH LISTIĆA

ALGORITHMS APPLICABLE IN THE PROCEDURE OF READING THE WORKING ACTIVITIES FROM TACHOCHARTS

Mikac M.¹

¹Veleučilište u Varaždinu, Varaždin, Hrvatska

Sažetak: Ključne informacije koje se vežu uz sigurnost cestovnog prijevoza robe i putnika vezane su uz tzv. radne aktivnosti vozača – podatke o trajanju vožnje, odmora i drugih zakonski definiranih aktivnosti tijekom radnog vremena. Praćenje radnih aktivnosti vozača bitna je stavka u logistici autoprijevozničkih tvrtki – osim kvalitetnijeg evidentiranja radnog vremena ti podaci služe i za otkrivanje eventualnih prekršaja i nepravilnosti u radu vozača. Za novoregistrirana vozila za prijevoz robe i putnika u RH vrijedi zakonska obveza ugradnje digitalnih tahografa. No, obzirom na prosječnu starost voznog parka autoprijevozničkih tvrtki u RH, još uvijek dominiraju ugrađeni analogni tahografi, koji podatke o radnim aktivnostima zapisuju na papirnate tahografske listiće. Preduvjet bilo kakve svrsishodne evidencije radnih aktivnosti jest digitalizacija tih podataka, odnosno očitavanje ključnih informacija s listića i prijenos na računalo. Ovim člankom predlaže se postupak za obradu i očitavanje podataka s tahografskih listića, opisuju algoritmi koje je moguće primijeniti u tom postupku i ukratko prezentira prateće programsko rješenje u sklopu kojeg su predloženi algoritmi implementirani.

Ključne riječi: analogni tahograf, radne aktivnosti vozača, digitalizacija tahografskih listića, očitavanje tahografskih listića, skeniranje, evidencija

Abstract: An important issue in the logistics of transport companies is efficient tracking of driver activities. It is directly related to public transport safety regulations, and therefore all companies have the obligation to provide required information. Since fleets of many Croatian transport companies include older vehicles equipped with analogue tachographs, digitalization of analogue tachocharts becomes highly important. This paper presents a process for tachochart digitalization and suggests few algorithms that were used in our analogue tachochart digitalization tool.

Key words: analogue tachograph, driver activities, tachograph chart, tachochart digitalization

1. UVOD

Profesionalni vozači i mobilni radnici obavljaju određene radne aktivnosti za vrijeme svog radnog vremena.

Zakonom [1] i važećim Pravilnikom [2] definirane su četiri vrste radnih aktivnosti – vožnja, spremnost za rad, ostali rad i odmor. Podaci o radnim aktivnostima bilježe se korištenjem posebnih uređaja ugrađenih u vozila – tahografima. Postoje dvije vrste tahografa – analogni i digitalni. Stara generacija uređaja, tzv. analogni tahografi, za pohranu informacije o vožnji i drugim aktivnostima koristila je analogni zapis, pri čemu je na kružni papirnati listić iscertavan status vozača. Nova generacija uređaja, koja je obavezna u svim novoregistriranim vozilima za prijevoz roba i putnika, potpuno je digitalna i podatke pohranjuje na digitalne pametne kartice koje se dodjeljuju vozačima i internu memoriju, što olakšava i ubrzava obradu potrebnih informacija.

Zakonski je propisano da autoprijevoznici pohranjuju i evidentiraju radne aktivnosti snimljene tahografima, pa se za te potrebe nastoji pronaći rješenje koje će evidenciju učiniti što jednostavnijom i efikasnijom.

Budući da je prosječna starost voznog parka autoprijevozničkih tvrtki u RH (a i u većini drugih tranzicijskih zemalja) takva da veliki broj (starijih) vozila ima ugrađene analogne tahografe, u praksi se pokazalo da se najviše vremena troši upravo na očitavanje i uvođenje u evidenciju podataka s tahografskih listića.



Slika 1. Analogni tahograf

Ovim člankom predlaže se efikasan i provjeren postupak za očitavanje podataka o radnim aktivnostima sa skeniranih listića. U prvom poglavlju opisuju se analogni tahografi, tahografski listići i princip bilježenja informacija o radnim aktivnostima na tahografske listiće. Drugo poglavlje daje pregled mogućih i dostupnih

rješenja za očitavanje podataka s tahografskih listića, pri čemu se posebno naglašava pristup skeniranja listića i ključne faze obrade prema predloženom postupku. U trećem se poglavlju sistematizira predloženi postupak, a ključni algoritmi se razmatraju odvojeno u sklopu četvrtog poglavlja. Peto poglavlje donosi programski sustav koji među ostalim implementira i cjelokupni izneseni postupak i algoritme.

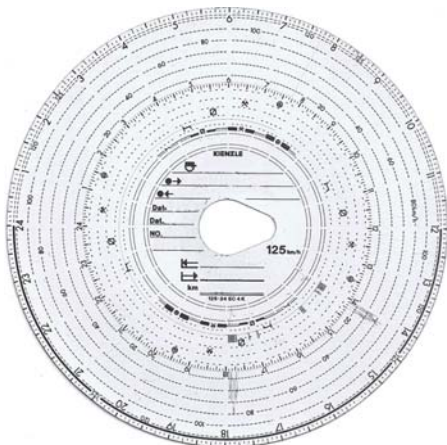
2. ANALOGNI TAHOGRAF I LISTIĆI

Analogni tahografi su starija generacija uređaja koja se ugrađivala u vozila za prijevoz robe i putnika. Na slici 1. prikazan je analogni tahograf. To je uređaj ugrađen u komandnu ploču vozila koji osim osnovnih pokazivača brzine ima ugrađene tipke za podešavanje radnih aktivnosti. Postoje i drukčiji modeli analognih tahografa, kao npr. na slici 2. – samostalni uređaj naknadno ugrađen u komandnu ploču.



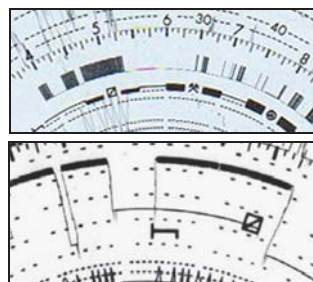
Slika 2. Analogni tahograf

Podatke o radu vozača analogni tahografi pohranjuju na tahografske listiće – papirnatu obrascu kružnog oblika promjera 12.3 cm. Izgled praznog, nekorištenog listića prikazuje slika 3. To je najčešće korišten tip listića u Hrvatskoj, s tim da su u novije vrijeme zbog očite potrebe za skeniranjem i digitalizacijom listića proizvođači ponudili i listiće sa čistim središnjim dijelom (slika 5.) za bilježenje aktivnosti (u kružnom vijencu u kojem se bilježe aktivnosti nema otisnutih simbola i drugih elemenata koji mogu utjecati na očitavanje). Oba tipa listića mogu se očitati predloženim algoritmom, s tim da se kod listića previdenih za digitalizaciju ne pojavljuju pogreške očitavanja uslijed otisnutih simbola.



Slika 3. Standardni taholistić

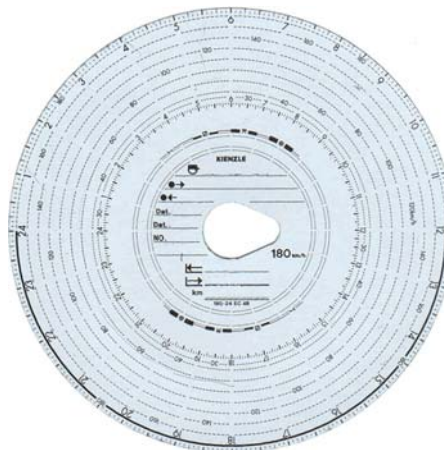
Postoje dva osnovna tipa zapisa aktivnosti na listiće – stariji analogni tahografi podatke o aktivnosti vozača zapisuju "stepeničasto", dok noviji (velika većina vozila s analognim tahografima u Republici Hrvatskoj) zapisuju podatke bez pomaka, unutar kružnog vijenca (slika 3.).



Slika 3. Standardni i stepeničasti zapis aktivnosti

Kod stepeničastog bilježenja aktivnosti (donji dio slike) različite aktivnosti se bilježe na različitim udaljenostima od središta listića. S druge strane, standardni zapis radnih aktivnosti ostvaren je na istoj udaljenosti od središta listića, ali se svaka aktivnost ispisuje različitom debljinom linije (gornji dio slike). Algoritam koji se predlaže ovim radom primjenjuje se isključivo na standardnom obliku zapisa.

Kao što je već spomenuto, podaci o radnim aktivnostima vozača zapisuju se unutar kružnog vijenca – podaci o prijeđenim kilometrima zapisuju se u dijelu listića bližem središtu, dok se na vanjskom dijelu listića bilježe podaci o trenutnoj brzini. Iako je i te podatke, teoretski, moguće očitati, postupak i algoritmi koji se opisuju u ovom članku ne obavljaju tu funkciju već su isključivo orijentirani na očitavanje radnih aktivnosti vozača.



Slika 5. Listić sa čistim kružnim vijencem

Obaveza vozača je prije umetanja listića da na predviđeno mjesto u središnjem dijelu ručno upiše svoje podatke, registarsku oznaku vozila i početno stanje kilometar sata. Po vađenju listića vozač mora upisati vrijeme (nije nužno, ali vrlo bitno kod noćnih vožnji kad vožnja počinje u jednom, a završava u drugom danu kako bi se očitani rezultati mogli ispravno interpretirati) i završno stanje kilometar sata.

Ispravno pozicioniranje listića unutar uređaja moguće je zbog asimetričnog središnjeg otvora. Osim što

omogućava ispravno ulaganje listića u uređaj, taj otvor je vrlo bitan i kod same obrade skeniranog zapisa jer pruža uvjete za određivanje orijentacije listića i detekciju početka vremenske skale na listiću.

3. OČITAVANJE LISTIĆA

Kao što je već spomenuto, zakonska je obaveza autoprijevoznika pohraniti i evidentirati podatke o radnim aktivnostima. Pohrana vezana uz pojedinog vozača obuhvaća arhiviranje tahografskih listića i preuzimanje podataka s digitalnih kartica vozača, dok vođenje evidencije podrazumijeva sistematsko praćenje radnih aktivnosti vozača. Preduvjet dobrog praćenja radnih aktivnosti je prijenos podataka u neki od dostupnih sustava ili programa za evidenciju.

Slijedom toga, podatke s tahografskih listića je nužno prenijeti na računalo u obliku koji će omogućiti daljnju obradu. Dva su osnovna pristupa – direktno očitavanje informacija s listića i očitavanje informacija sa skeniranih listića.

Direktno očitavanje listića može se odraditi ručno (vizualno, uz pomoć ravnala ili sličnih pomoćnih alata, se određuju granična vremena pojedinih aktivnosti i ti podaci se ručno unesu na računalo). Kod složenijih listića mogu se koristiti specijalizirani uređaji koji olakšavaju rad korisniku, no kod listića s mnogo čestih promjena aktivnosti uređaji su teže iskoristivi. Primjer specijaliziranog uređaja u ponudi tvrtke VDO [3] je prikazan na slici 6.

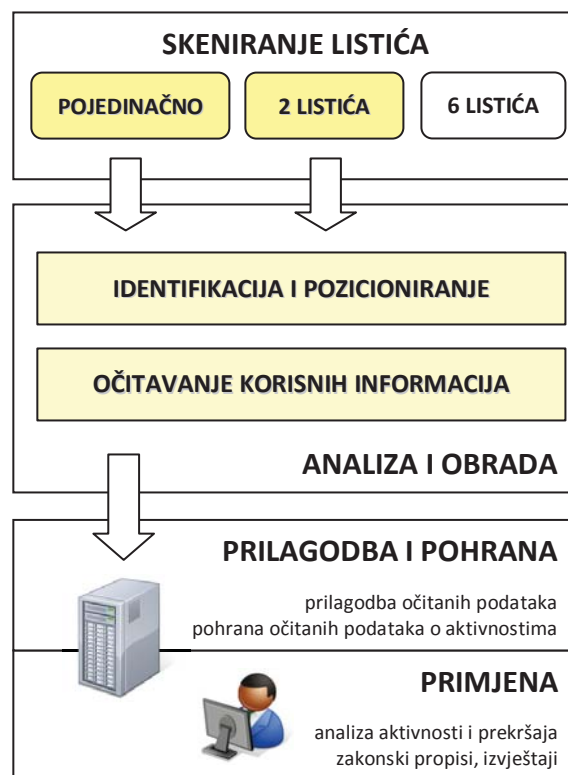


Slika 6. Specijalizirani uređaj za čitanje listića

Pristupom koji uključuje skeniranje listića, postupak digitalizacije se djelomično automatizira i obavlja u nekoliko koraka. Prvi korak je skeniranje listića, a rezultat je digitalni slikovni zapis na računalo. Drugi korak uključuje analizu slikovnog zapisa i identifikaciju te pozicioniranje listića. Nakon uspješne identifikacije listića provodi se automatsko očitavanje korisnih i traženih informacija. Završni korak je preuzimanje očitanih podataka, prilagodba i pohrana u evidencijski sustav i daljnja obrada. Dijagram postupka prikazuje slika 7.

3.1. Skeniranje listića

Skeniranje listića je postupak kojim se posebnim uređajem, skenerom, na računalo prenosi slikovni zapis listića. Alternativno, postupak je moguće provesti i korištenjem fotoaparata, odnosno fotografiranjem listića. Rezultat je slikovni zapis u nekom od standardnih slikovnih formata.



Slika 7. Postupak digitalizacije baziran na skeniranju

S obzirom na to da vozači moraju u vozilu čuvati listiće za posljednjih mjesec dana vožnje, u praksi se obrada podataka provodi s vremenskim odmakom i uključuje obradu većeg broja listića za pojedinog vozača. Jasno je da postupak skeniranja većeg broja listića može potrajati, pa stoga izbor uređaja za skeniranje direktno utječe na vrijeme koje je potrebno utrošiti za postupak skeniranja. Dodatno ubrzanje obrade može se postići i kombiniranim skeniranjem više listića odjednom (2 ili 3 potpuna listića na A4 skenerima, 6 djelomičnih listića – skeniranje samo središnjem kružnog vijenca s podacima o aktivnostima).

Standardni kućni skeneri omogućavaju skeniranje u zadovoljavajućoj kvaliteti i za skeniranje površine dimenzije A4 (ili nešto većih od A4, ovisno o modelu) potrebno im je 20-30 sekundi. Skeniraju li se po dva listića odjednom, dolazimo do podatka o 10-15 sekundi potrebnih za skeniranje pojedinog listića.



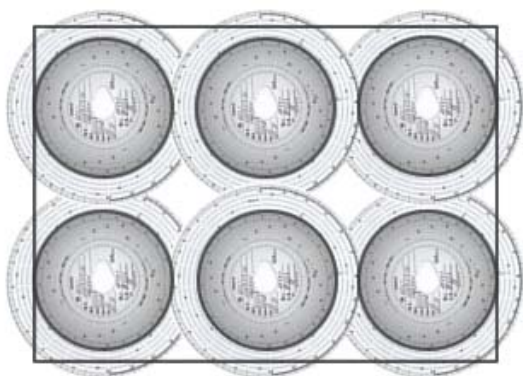
Slika 8. Profesionalni brzi skener listića

Profesionalni specijalizirani skeneri (slika 8.) mogu skeniranje obaviti mnogo brže. Proizvođač gornjeg skenera je Panasonic, a tvrtke specijalizirane za obradu podataka ga najčešće, uz prilagođen softver, prodaju pod

svojim imenom. Dostupne informacije o brzini variraju pa ih treba uzeti s rezervom (30-40 listića u minuti). Neke cijenom pristupačnije skenere primjenjive u postupku, a dostupnih na domaćem tržištu, nudi proizvođač Canon, s tim da je tijekom testiranja postignuta brzina očitavanja oko 15 listića u minuti.

Ključne informacije pohranjene na listiću su informacije o radnim aktivnostima vozača. S obzirom ne to da se ti podaci nalaze unutar kružnog vijenca u središnjem dijelu listića, za njihovo očitavanje nije nužno da skenirani listić bude cjelovit – taj pristup može se iskoristiti za plošno skeniranje većeg broja listića – do 6 listića na području standardnih A4 dimenzija. U tom slučaju je vrlo bitno precizno pozicionirati listiće - najbolje uz korištenje vodilica ili posebnih držača za listiće koji osim pozicije fiksira i orijentaciju listića.

Algoritam koji se iznosi u ovom članku može se primijeniti na takav oblik skeniranih listića uz minimalne preinake – ključno je ispravno pozicionirati listiće na skeniranom zapisu prije same obrade i očitavanja.



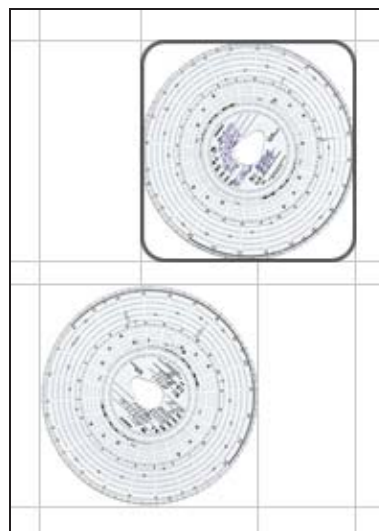
Slika 9. Skeniranje 6 necjelovitih listića na A4

Slika 9. prikazuje A4 list (položeno) i mogućnost pozicioniranja listića za djelomično skeniranje (necjeloviti listići). Prednost tog postupka je povećanje brzine skeniranja (6 listića za 20-30 sekundi pri korištenju standardnih plošnih skenera), no glavni nedostatak je da tako skenirani listići nisu potpuni pa se ne mogu koristiti za stvaranje digitalne arhive. Kod skeniranja pojedinačnih ili 2-3 listića na A4 moguće je snimke pohraniti i koristiti kao digitalnu arhivu jer sadrže kompletne listiće i omogućavaju naknadno očitavanje ostalih parametara, a ne samo radnih aktivnosti.

3.2. Identifikacija i pozicioniranje listića

Nakon što se nekim od opisanih postupaka provede skeniranje potrebno je na slikovnom zapisu identificirati i locirati listić – naći poziciju listića na samoj slici. Iako pogonski programi za skenere nude mogućnost automatskog rezanja i pronalaženja korisnog sadržaja na slikama, u praksi se pokazalo da softver najpoznatijih proizvođača kućnih skenera (Canon, HP) nije pouzdan u pronalaženju listića i često rezultira gubitkom podataka (vjerojatno zbog male razlike u intenzitetu podloge i listića).

Zbog toga se prije primjene algoritma za očitavanje radnih aktivnosti, neovisno o broju skeniranih listića po zapisu, mora izvesti algoritam za određivanje pozicija i identifikaciju listića. Kao dio alata za očitavanje taholistića implementiran je algoritam koji precizno pronalazi rubove i pozicije pojedinačno skeniranih listića i listića na skenovima koji sadrže dva listića. Slika 10. prikazuje slučaj dva skenirana listića – na slici su linijama označene granice koje algoritam za pozicioniranje određuje tijekom izvođenja.



Slika 10. Identifikacija listića na skeniranom zapisu

Algoritam za identifikaciju listića u svojoj općenitoj varijanti (moguća buduća implementacija podrške za veći broj listića) kao rezultat vraća broj listića i koordinate gornje lijeve i donje desne pozicije pravokutnog (kvadratnog) područja unutar kojeg je pojedini listić smješten.

3.3. Očitavanje korisnih informacija

Iako se sa slikovnog zapisa tahografskog listića, kako je već opisano u poglavlju 2., može očitati više vrsta informacija, algoritam koji se predlaže odnosi se isključivo na očitavanje radnih aktivnosti. Konačni rezultat očitavanja mora biti strukturno dan u obliku prikazanom tablicom 1.

Tablica 1. Struktura rezultata očitavanja radnih aktivnosti

Tip aktivnosti	Simbol	Od	Do
Odmor	↪	05:15*	06:24*
Vožnja	⊙	*početak i kraj razdoblja unutar kojeg je obavljena određena radna aktivnosti	
Spremnost	☑		
Ostali rad	⚡		

Sasvim je jasno da prilikom očitavanja algoritam mora odrediti o kojem se tipu aktivnosti radi – određivanje tipa aktivnosti temelji se na širini (debljini) zapisa na određenoj poziciji. Pravilo za bilježenje radnih aktivnosti

definira da se vožnja bilježi najdebljom linijom, a odmor najtanjom linijom, što prikazuje tablica 2.

Tablica 2. Standardni zapis radnih aktivnosti na listiću

Tip aktivnosti	Simbol	Zapis
Odmor		
Vožnja		
Spremnost		
Ostali rad		

Iako su svi tipovi aktivnosti važni, većina zakonski propisanih pravila definira uvjete vezane uz odmor i vožnju. S obzirom na to da su te aktivnosti na popunjenom listiću vizualno različite, očitavanje upravo tih tipova aktivnosti postiže se s vrlo visokom točnošću, dok kod očitavanja spremnosti i ostalog rada može doći do manjih nepravilnosti koje ponekad operater mora ručno korigirati prije pohrane u evidencijski sustav.

Za određivanje početka i kraja razdoblja obavljanja pojedine aktivnosti treba za određenu poziciju na kružnom vijencu odrediti o kojem se vremenu unutar dana radi. S obzirom na to da je riječ o kružnom zapisu, 24-satno razdoblje (dan) bilježi se unutar 360°. Drugim riječima, unutar jednog stupnja mogu se prikazati 4 minute radne aktivnosti.

Očitavanje radnih aktivnosti provodi se direktno iz slikovnog zapisa – za potrebe očitavanja se kružni vijenac linearizira i pretvara u pravokutnik. Budući da ni zapisi o aktivnostima na modernim digitalnim tahografima ne osiguravaju preciznost očitavanja ispod 1 minute [5], vijenac se linearizira u pravokutnik širine 1440 točaka (piksela). U tom slučaju svaka točka (linija) lineariziranog zapisa predstavlja minutu. Princip prikazuje slika 11. (prilagođeni prikaz iz programskog alata u kojem su implementirani opisivani algoritmi). Na listiću su tri pozicije (A, B i C) koje su označene i na

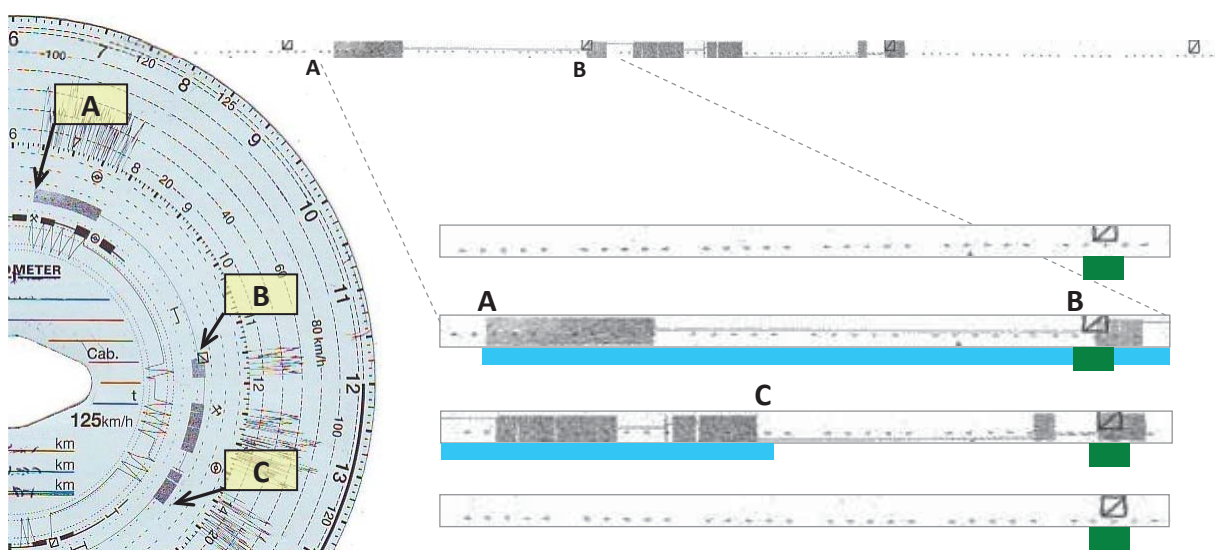
uvećanom lineariziranom prikazu. Dio očitavanja od A do C posebno je naznačen ispod prikaza. Uz poziciju B ukazuje se na pojavu simbola spremnosti za rad koja može uzrokovati pogrešna očitavanja u kasnijoj fazi obrade. Simbol se pojavljuje četiri puta unutar cijelog kružnog vijenca. Na listićima poput onog prikazanog na slici 5., taj problem se ne pojavljuje jer simboli nisu otisnuti unutar radnog područja.

Iako pitanje preciznosti očitavanja ovisi o kvaliteti skeniranih zapisa, u praktičnoj primjeni algoritama minutne preciznosti pokazalo se da su dovoljne dimenzije slikovnog zapisa pojedinog listića 1023x1023 točaka. Kako objasniti te minimalne dimenzije? Za linearizaciju kružnog vijenca u pravokutni širine 1440 piksela nužno je da za polumjer unutarnje kružnice vrijedi:

$$r \geq \frac{1440}{2 \cdot \pi}, \quad r \geq 230 \text{ pixel}$$

Budući da pozicije na koje uređaji isrtavaju aktivnosti mogu varirati od modela do modela, a obrasci listića su standardizirani, za određivanje minimalne zadovoljavajuće veličine listića možemo kao uvjet postaviti da polumjer područja unutar kojeg je dopušteno isrtavanje aktivnosti bude najmanje 230 točaka. Kako je ta pozicija precizno definirana i standardizirana i nalazi se na 0.45 polumjera listića, možemo izračunati da minimalni polumjer listića mora biti 511 točaka, odnosno dimenzije listića koje će omogućiti minutnu preciznost očitavanja moraju biti bar 1022x1022 točaka. Uzmemo li u obzir kružni oblik listića, za precizno određivanje središta povećavamo dimenziju za jedan piksel – neparan broj točaka omogućava određivanje središta bez pomaka.

U usporedbi s drugim pristupima očitavanju listića (varijanta bez skeniranja), minutna preciznost je velika prednost. Međutim, u praksi aktivnosti koje traju ispod 5 minuta najčešće se zanemaruju ili spajaju sa susjednim aktivnostima, pa i rezultati dobiveni obradom skeniranih listića dimenzija manjih od predloženih posjeduju zadovoljavajuću razinu preciznosti.



Slika 11. Primjer linearizacije dijela kružnog vijenca

Tablica 3. prikazuje odnos preciznosti i okvirnih dimenzija skeniranih listića.

Dimenzije	Preciznost	Direktno očitavanje
1022x1022	1 min	
900x900	1.13 min	1272 min.
800x800	1.27 min	1130 min.
700x700	1.45 min	989 min.
400x400	2.54 min	565 min.

Tablica 3. Dimenzije listića i preciznost očitavanja

Očito je da sve spomenute dimenzije listića iz tablice mogu zadovoljiti uvjet vrlo dobre preciznosti. Daljnje smanjenje listića matematički bi i dalje zadovoljavalo što se tiče preciznosti (ispod 5 minuta), no radilo bi se o gotovo nečitljivim listićima i primjena ne bi imala smisla. Testiranja algoritma su pokazala da se iz slikovnih zapisa niže rezolucije (npr. listići fotografirani mobitelom) može također provesti dovoljno dobro očitavanje. No, za profesionalno korištenje i paralelno vođenje digitalne arhive preporuka je korištenje zapisa skeniranih s minimalno 150 dpi (točaka po inču) – dimenzije listića kod takvog skeniranja su oko 730 piksela što je dovoljno dobro i nudi preciznost očitavanja ispod 1.5 minute.

Za potrebe očitavanja drugih informacija s listića, podataka o prijeđenim kilometrima iz središnjeg dijela listića (manji polumjeri), uvjeti o kvaliteti zapisa postaju stroži. Kao što je već spomenuto, prezentirani algoritmi su primarno razvijani s ciljem pouzdanog očitavanja radnih aktivnosti, pa prije navedeni uvjeti zadovoljavaju taj slučaj primjene.

Nakon provedene linearizacije provodi se očitavanje aktivnosti. Za svaku minutu (svaki stupac točaka) očitava se vrijednost snimljenih točaka (intenzitet, s obzirom na to da je riječ o tamnim bilješkama na svijetloj podlozi); prebrajaju se tamni pikseli te se obavlja kategorizacija – dominacija tamnih piksela zasigurno znači da se radi o vožnji, dok minimalna prisutnost tamnih piksela signalizira da se radi o odmoru ili neradnom dijelu dana (što u konačnici sustav evidentira kao odmor). Preostale dvije aktivnosti se preciznije obrađuju i provodi se kategorizacija.

Postoje neki slučajevi u kojima dolazi do pogrešnih očitavanja, a to su situacije u kojima se radi o oštećenim listićima, problemi s listićima koji nisu prilagođeni za digitalnu obradu, sjene koje nastaju kod skeniranja i sl. Tako kod listića prikazanih na slici 4., za razliku od listića sa čistim kružnim vijencem prikazanim na slici 5., na određenim pozicijama može doći do preklapanja odštampanih simbola i snimljenih aktivnosti (slika 11.), što ponekad dovodi do pogrešnih očitavanja. Iako autori u [6] i [7] spominju mogućnost eliminacije tih problema zbog razlika u intenzitetima originalno otisnutih oznaka i samih tahografskih snimaka, ta pravilnost u korištenim primjerima nije identificirana niti je spomenuti problem uspješno riješen (eliminiran). I u slučaju precizne

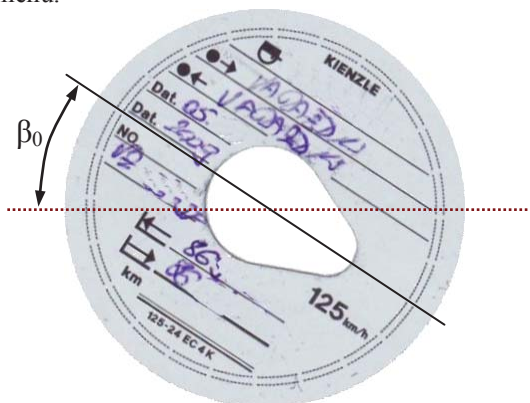
identifikacije odštampanih simbola ostaje problem odlučivanja i interpolacije – što je tahograf na toj poziciji zapisao? Utjecaj tog problema dodatno umanjuje svijest korisnika tahografa o potrebi korištenja listića prilagođenih digitalnoj obradi. Ukratko, problem pogrešnih očitavanja zbog otisnutih simbola predloženim algoritmima nije rješavan.

Neovisno o tome, rezultati očitavanja i u slučaju problema su zadovoljavajući, s obzirom na to da na razini cijelog dana tijekom testiranja maksimalni broj pogrešnih očitavanja varira unutar 5-10 minuta – što je pogreška ispod 1%. Ovisno o očitanim aktivnostima, pogreške su minimalne, gotovo zanemarive kod očitavanja vožnje i odmora, što zapravo predstavlja ostvarenje definiranog cilja.

3.4. Dodatna obrada i prilagodba očitavanja

Očitavanja dobivena iz lineariziranog prikaza kružnog područja s oznakama aktivnosti omogućavaju određivanje traženih vremenskih granica za pojedinu aktivnost. Međutim, dobivena vremena zapravo pokazuju samo relativan odnos prema nekom početnom vremenu – počne li linearizacija na 0° slikovnog zapisa (horizontalno), sva će očitana vremena pokazivati pomak u odnosu na tu poziciju.

Postavlja se pitanje kako odrediti apsolutno točno vrijeme, odnosno točno vrijeme unutar dana kada je aktivnost zabilježena? Vizualnim očitanjem na listiću je lako očitati točno vrijeme, s obzirom na to da je na obrascu ispisana vremenska skala. Prilikom automatizirane obrade za točno očitavanje se koristi specifičnost listića – asimetrična središnja šupljina. Naime, izduženje središnjeg otvora može se iskoristiti za određivanje referentnog vremena – izduženi dio uvijek pokazuje na 12 sati, nasuprot toj poziciji nalazi se početak/kraj dana, odnosno 24 sata. Kod transformiranja i očitavanja riječ je o čistom matematičkom postupku translacije vrijednosti, nakon što se odredi referentna točka (kut zakrenutosti β_0 listića u odnosu na očekivanu orijentiranost - horizont), nije problem ostvariti pomak po vremenskoj osi i dobiti očitavanja u apsolutnom vremenu.



Slika 12. Orijentacija listića – središnja šupljina

U praksi se ipak pokazalo da određivanje kuta zakrenutosti β_0 temeljeno na analizi središnje šupljine nije uvijek precizno. Odstupanje od samo jednog stupnja može uzrokovati pogreške u vremenskom pozicioniranju

Na dijagramu postupka (slika 14.) posebno su označeni funkcionalni blokovi unutar kojih su implementirani algoritmi prezentirani ovim člankom, a naznačene su i mogućnosti intervencije korisnika. U slučaju intervencije, slijedno se provode sve daljnje funkcije i u realnom vremenu prikazuju novi rezultati očitavanja.

Očitani podaci se korisniku prezentiraju tablično u zahtijevanom obliku kako je već prikazano tablicom 1.

Osim direktnog očitavanja aktivnosti, u postupku je moguće aktivirati i naknadnu prilagodbu očitanih podataka. Npr., korisnik može zahtijevati da se sve kratke aktivnosti (npr. ispod 2 minute) preskoče ili povežu sa susjednim aktivnostima i sl. pa ti zahtjevi konačno utječu na rezultate očitavanja i korisnik ih koristi na vlastitu odgovornost upravo zbog tog malog, ali ipak stvarnog utjecaja na očitavanja.

5. ALGORITMI PRIMJENJIVI U POSTUPKU

Provođenje digitalizacije i očitavanja informacija s tahografskog listića uključuje korištenje više algoritama raznih namjena. U ovom poglavlju ukratko su opisani algoritmi primjenjivi u postupku – implementacijom predloženih algoritama stvoren je temelj za razvoj programskog rješenja za očitavanje radnih aktivnosti.

Postupak uvijek počinje primjenom algoritama za pronalazak listića - I a) - `algDetectAnalogTachoCharts`. Algoritam se opisuje u odjeljku 5.1. Rezultat tog algoritma je grubo lociranje listića (jednog ili dva) na slikovnom zapisu, nakon čega se primjenjuje algoritam za određivanje pozicije listića na skeniranom zapisu - I b) - `algTachoChartPosition`. Algoritam je opisan u odjeljku 5.2.

Očekivano središte listića određuje se prema očitanim dimenzijama (uz mogućnosti korisničkog postavljanja). S obzirom na definirano središte, primjenjuje se algoritam (odjeljak 5.3.) za određivanje orijentacije, odnosno zakrenutosti listića - I d) - `algTachoChartOrientation`. Na sličan način (automatizmom prema standardnim dimenzijama listića ili na temelju korisničkog unosa) određuju se rubovi kružnog isječka s podacima o aktivnostima. Prema tako definiranim parametrima (polumjer vanjskog i unutarnjeg kruga) može se napraviti linearizacija zapisa o aktivnostima primjenom algoritma `algLinearizeTachoChart` – II a).

U opisima algoritama daju se osnovne značajke algoritma, definiraju se ulazni i kontrolni parametri i rezultati, odnosno izlazni parametri. Uz algoritme se spominje opisni pseudo-kod.

5.1. Algoritam pronalaska listića

Algoritam za pronalazak listića na skeniranom slikovnom zapisu, `algDetectAnalogTachoCharts`, zapravo je standardni *trim/crop* algoritam kojem je zadatak na podlozi identificirati jedan ili dva listića i otprilike odrediti njihove pozicije i dimenzije na originalnom zapisu.

Algoritam:	<code>algDetectAnalogTachoCharts</code>
Ulaz:	skenirani zapis
Izlaz:	broj pronađenih listića (0, 1 ili 2) pozicije i dimenzije listića
Kontrola:	granični intenzitet D

Postupak se svodi na pretraživanje ulaznog zapisa - redom se analiziraju (horizontalno i vertikalno) točke ulaznog zapisa i traže se nagle promjene intenziteta.

S obzirom na to da veličine ulaznih slikovnih zapisa direktno utječu na vrijeme izvođenja algoritma, s ciljem brže obrade, prije početka analize ulazni zapis se smanjuje tako da se manja dimenzija reducira na 400 točaka, a druga dimenzija reducira razmjerno tome. Jasno je da se ovim pristupom gubi na kvaliteti određivanja pozicija listića, no to nije ključno budući da se u sljedećem koraku obavlja detaljno lociranje rubova listića. Korisna nuspojava ovakvog pristupa je eliminacija sitnih smetnji koje se mogu javiti na pozadini originalnog zapisa, s obzirom na umanjenje. Manje smetnje gotovo nestaju i samim tim se reducira njihov utjecaj na pronalazak listića.

Pseudo-kod algoritma za detekciju sadržaja na podlozi:

```

1. Učitaj originalni slikovni zapis
2. Odredi potrebni faktor umanjenja  $m$ 
3. Umanji originalni zapis za faktor  $m$  - dimenzije ( $mX, mY$ )

4. Pronađi vertikalne rubove sadržaja
   (moguć paralelni rad ili izvršenje u istoj petlji...)

   Za svaki  $i$  ( $1..mY$ ) analiziraj redak
   {
       Za svaku točku ( $i, 1..mX$ ) provjeri okolinu dolje

       Promjena intenziteta  $> D \rightarrow$  gornji rub  $tY=i$ 
   }
   Za svaki  $i$  ( $mY..1$ ) analiziraj redak
   {
       Za svaku točku ( $i, 1..mX$ ) provjeri okolinu gore

       Promjena intenziteta  $> D \rightarrow$  donji rub  $bY=i$ 
   }

5. Pronađi horizontalne rubove sadržaja
   Analiza u smanjenom području ( $tY, bY$ )

   Za svaki  $i$  ( $1..mX$ ) analiziraj stupac
   {
       Za svaku točku ( $i, tY..bY$ ) provjeri okolinu desno

       Promjena intenziteta  $> D \rightarrow$  lijevi rub  $lX=i$ 
   }
   Za svaki  $i$  ( $mX..1$ ) analiziraj stupac
   {
       Za svaku točku ( $i, tY..bY$ ) provjeri okolinu lijevo

       Promjena intenziteta  $> D \rightarrow$  desni rub  $rX=i$ 
   }

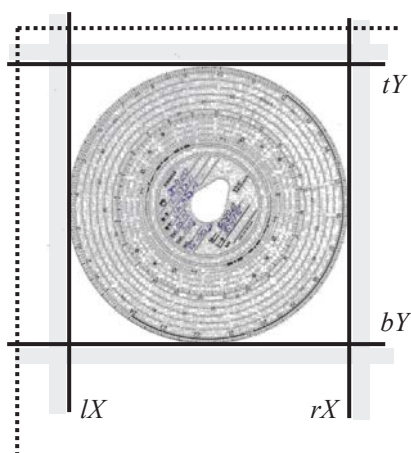
6. Skaliraj rubne vrijednosti na originalan zapis
   uz dodavanje "sigurnosne granice"

7. S obzirom na umanjenje u (3) dobivene vrijednosti imaju
   umanjenju preciznost, no dobro su početno rješenje
   za nastavak postupka

```

Opisani algoritam kao rezultat vraća rubne parametre (lX, tY, rX, bY) koji zapravo određuju poziciju gornje lijeve i donje desne rubne točke pravokutnika koji opisuje listić (slika 15.). Točka 6. algoritma namjerno ugrađuje "pogrešku", odnosno proširuje rezultatno područje. Na taj način smanjuje se utjecaj umanjenja originalnog

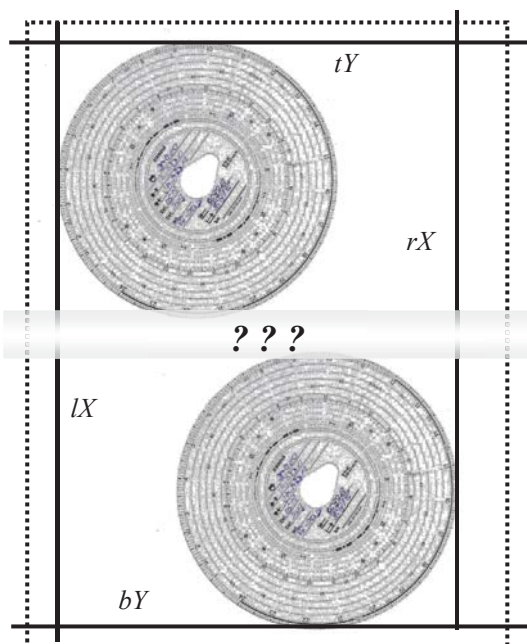
zapisa i moguće pogreške očitavanja. Na slici je iscrtkano naznačen rub skeniranog zapisa, punim crnim crtama su označeni izlazni parametri – rubovi listića, dok sivo područje označava proširenje područja koje se namjerno ubacuje zbog skaliranja slike u prvim koracima algoritma.



Slika 15. Rezultati algoritma na jednom listiću

U slučaju da skenirani zapis sadrži dva listića, jasno je da rezultat neće biti korektan, odnosno postoji potreba za dodatnom analizom - slika 16. Na slici su označeni rubni parametri dobiveni algoritmom, a posebno je istaknuto središnje područje koje vizualno odjeljuje listiće i koje bi unaprijeđeni algoritam trebao izolirati i uzeti u obzir u daljnjoj obradi.

Ideja je iskoristiti dobivene vanjske rubove sadržaja i daljnjom analizom odrediti postojanje "praznih" redova – ako takvi postoje, to znači da se na skenu nalazi više razdvojenih objekata.



Slika 16. Rezultati algoritma – dva listića

Dodatak algoritmu pseudo-kodom može se prikazati na sljedeći način:

```

1. Pripremi analizu prema rubnim (lX,tY,rX,bY), Y0=Y1=-1
2. Pronađi "praznu horizontalnu traku"
   Za svaki i (tY..bY) - redak
   {
       ako je prazan redak
       {
           ako je Y0=-1 zabilježi prvi Y0=i
           inače Y1=i
       } inače
       {
           ako je Y0<>-1 prekid
       }
   }

3. Ako je Y0<>-1 imamo više od jednog listića
   {
       odredili smo rubove praznog područja
       podijeli vertikalno tY-Y0 i Y1-bY
       obradi i pronađi lX i rX za oba dijela slike
       → (prilagođena prva faza s ulaznim područjima)
   } inače
   {
       radi se o jednom listiću, ostaju (lX,tY,rX,bY)
       daljnja obrada nije potrebna
   }

```

Potreba za ovim algoritmom rezultat je želje da se korisnicima omogući "istovremena" obrada dva skenirana listića, bez korištenja pomoćnih alata za fiksiranje pozicija prilikom skeniranja i sličnih rješenja. Algoritam nije optimiziran niti se primjenjuju bilo kakvi napredni mehanizmi za otkrivanje većeg broja listića. Može se pronaći jedan ili dva listića na radnoj površini i to u slučaju da između njih postoje prazni horizontalni dijelovi slike – dva listića skenirana tako da su pozicionirani jedan ispod drugog. Ako se otkrije prazno horizontalno područje unutar granica dobivenih prvim dijelom algoritma, može se zaključiti o postojanju dva ili više listića na skeniranom zapisu.

5.2. Algoritam za određivanje pozicije listića

Ideja algoritma za određivanje pozicije listića je identična prethodno opisanom algoritmu za pronalazak listića na skeniranom zapisu.

Algoritam:	algTachoChartPosition
Ulaz:	skenirani zapis neprecizna pozicija i dimenzije (5.1)
Izlaz:	precizna pozicija i dimenzije listića
Kontrola:	granični intenzitet, osjetljivost

Ulazni parametri algoritma uključuju grubo određenu poziciju listića na skeniranom zapisu i grubo određene dimenzije listića. To su zapravo izlazni parametri algoritma iz 5.1.

Ideja preciznog određivanja pozicije listića svodi se na pretraživanje unutar grubo određenih granica. Tako se smanjuje područje pretraživanja, što je zapravo jedini način ubrzanja postupka. Umjesto pretrage cijelog ulaznog zapisa pretražuje se samo dio. Pseudo-kod algoritma je identičan onom iznesenom uz algoritam iz 5.1., a jedina je razlika preskakanje već opisanog namjernog proširenja područja u točki 6.

Pomoćna transformacija za detekciju rubova

S obzirom na to da se kod skeniranih zapisa radi o svijetloj pozadini i tamnim oznakama rubova i sadržaja na skeniranom listiću (sam listić je također svjetliji), jasno je da određivanje rubova područja listića nije veliki problem. Međutim, prilikom skeniranja se javljaju problemi sjena na rubnim područjima, a moguće su i druge smetnje i pogreške prilikom skeniranja koje mogu dovesti do krivih očitavanja i težeg razlikovanja listića od pozadine. Iako se korektna rješenja mogu u određenim slučajevima dobiti i analizom originalnih zapisa, nad originalnim zapisom provodi se transformacija kojoj je cilj dodatno izolirati rubna područja, odnosno područja većih promjena intenziteta svjetlosti. Za potrebe detekcije rubova korišten je Sobelov operator. U postupku odabira adekvatne metode detekcije testirani su i Scharrov i Cannyev operator. Ostvarenje dobrih rezultata i jednostavnost implementacije doveli su do odluke o korištenju Sobelovog operatora kao transformacijskog postupka za određivanje rubova. Sobelov operator u dvodimenzionalnim sustavima definira se kao:

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} * A \quad G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} * A$$

$$G = \sqrt{G_x^2 + G_y^2}$$

Transformacija se provodi nad željenim zapisom ili dijelom zapisa – ulaznom matricom A , a to je matrica dimenzija jednakih dimenzijama područja koje se transformira. Elementi matrice su vrijednosti intenziteta pojedine točke slikovnog zapisa.

Rezultat je matrica G - slikovni zapis s naglašenim razlikama intenziteta na mjestima gdje su detektirani nagli prelasci (rubovi, engl. *edge*) intenziteta originalnih zapisa.

Nakon transformacije pretraživanje slikovnog zapisa i pronalaženje rubova je jednostavno: kretanjem od rubova područja pretraživanja ispituju se transformirane točke i detektiraju se područja znatnih promjena intenziteta. Osjetljivost (minimalna razlika intenziteta) je jedan od parametara samog algoritma, a u konkretnom alatu za očitavanje listića definirane su inicijalne vrijednosti koje korisnik može po želji mijenjati ako se za to ukaže potreba. Algoritam za transformaciju je prilagođen na način da zanemaruje točke izvan područja tražene osjetljivosti.

S obzirom na to da transformacija više slikovnih zapisa velikih dimenzija može biti procesorski zahtjevna i može oduzeti dosta vremena, u ostvarenju očitavanja aktivnosti broj transformacija i transformirano područje su maksimalno reducirani - algoritam 5.2. i 5.4. izvršavaju se korištenjem zajedničkog transformata koji se jednokratno provodi prilikom izvođenja algoritma 5.2.

5.3. Algoritam za određivanje orijentacije listića

U odjeljku 3.4. opisana je ideja određivanja orijentacije listića. Algoritam se odvija u dvije faze: u početnoj fazi analizira se središnja šupljina listića te se određuje

početni kut zakrenutosti β_0 , dok se u drugoj fazi iskorištava karakteristični izgled ruba vremenske skale na listiću i preciznije određuje kut β .

Algoritam:	algTachoChartOrientation
Ulaz:	slikovni zapis, koordinate središta
Izlaz:	kut β

Prva implementacija u alatu za očitavanje aktivnosti vozača uključivala je samo prvu fazu algoritma. S obzirom na dostupna rješenja, već i približno određivanje kuta (β_0) na skeniranom zapisu (uz mogućnost ručnog podešavanja) korisnicima je olakšavalo i ubrzavalo rad. Implementacijom dvofazne varijante algoritma pokazalo se da uvođenje dodatne analize rubnog područja listića u gotovo svim testnim uzorcima očitavanje kuta β obavlja vrlo precizno i bez ikakvih odstupanja.

Pseudo-kod algoritma za detekciju početnog kuta β_0 :

<ol style="list-style-type: none"> 1. Učitaj originalni slikovni zapis 2. Za svaki kut k ($0..360$) s pomakom 0.25 (1440 koraka) <ul style="list-style-type: none"> { <li style="padding-left: 20px;"><i>kreni od središta listića</i> <li style="padding-left: 20px;"><i>pomiči se od središta do tamnog područja</i> <li style="padding-left: 20px;"><i>zapamti broj točaka do granice (interpoliranje)</i> } 3. Odredi kutove s najvećim brojem točaka 4. Dodaj kutove s manjim odstupanjima 5. Prema simetriji odredi središnji kut β_0
--

Pseudo-kod algoritma za preciznije određivanje kuta β – druga faza algoritma, kao ulaz prima kut β_0 :

<ol style="list-style-type: none"> 1. Odredi područje pretraživanja (<i>udaljenosti vremenske skale od središta</i>) 2. Za svaki kut k ($\beta_0 - X.. \beta_0 + X$) s pomakom 0.25 <ul style="list-style-type: none"> { <li style="padding-left: 20px;"><i>prebroji tamne točke unutar područja pretrage</i> } 3. Pronađi kut s najmanjim brojem točaka ili s najvećom promjenom broja točaka u odnosu na susjede → β

5.4. Algoritam za linearizaciju zapisa

Linearizacija kružnog vijenca koji sadrži podatke o radnim aktivnostima obavlja se jednostavnim postupkom transformacije opisanom u odjeljku 3.3.

Algoritam:	algLinearizeTachoChart
Ulaz:	slikovni zapis A , koordinate središta (X_S, Y_S), polumjeri radnog područja (R_1, R_2)
Izlaz:	linearizirani slikovni zapis kružnog vijenca B

Da bi se linearizacija mogla provesti, u algoritam se osim originalnog slikovnog zapisa proslijeđuju koordinate središta listića i polumjeri radnog područja. To su vrijednosti koje se automatski određuju, ali se i korisniku nudi mogućnost vizualnog podešavanja.

Rezultat algoritma je slikovni zapis, odnosno matrica intenziteta širine 1440 točaka, pri čemu svaki stupac predstavlja očitavu aktivnost u određenoj minuti.

Pseudo-kod algoritma za linearizaciju:

```

1. Odredi područje pretraživanja (rubovi kružnog vijenca)
2. Za svaki kut  $k$  ( $0..360$ ) s pomakom  $0.25$  (1440 koraka - minuta)
   {
   za svaki  $r$  ( $R1..R2$ )
   {
       očitaj točku na slikovnom zapisu
        $A(XS+r\cos(k), YS+r\sin(k))$ 

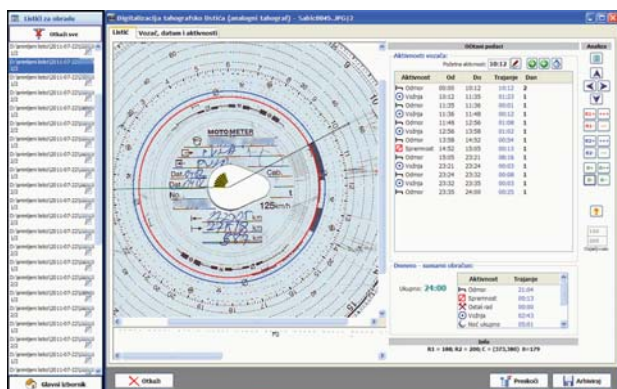
       pohrani boju/intenzitet u  $B(4*k, r-R1)$ 
   }
   }
3. Vrati slikovni zapis  $B$ 

```

U ovisnosti o kutu zakrenutosti listića β u sklopu iz rezultantnog zapisa/matrice B lako se dobiva očitavanje usklađeno s vremenskom skalom listića.

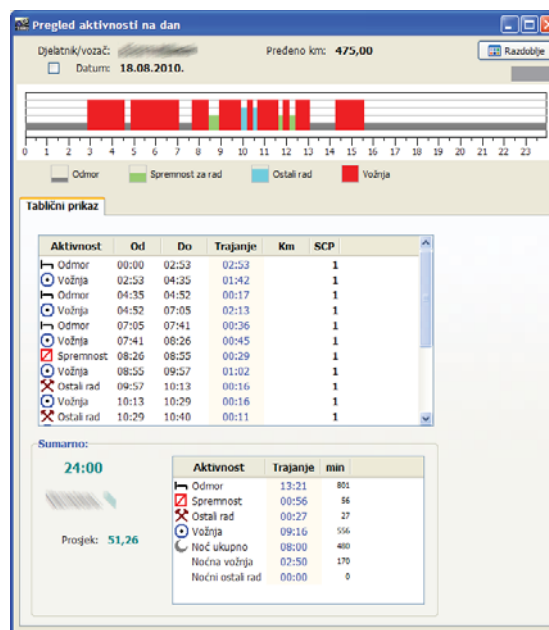
6. PROGRAM ZA OČITANJE I INTERPRETACIJU ZAPISA

Postupak za očitavanje skeniranih tahografskih listića iznesen u ovom članku potpuno je implementiran kroz programsko rješenje razvijeno za Windows operacijske sustave. Primjer korisničkog sučelja za očitavanje i manipulaciju očitanim podacima prikazuje slika 17.

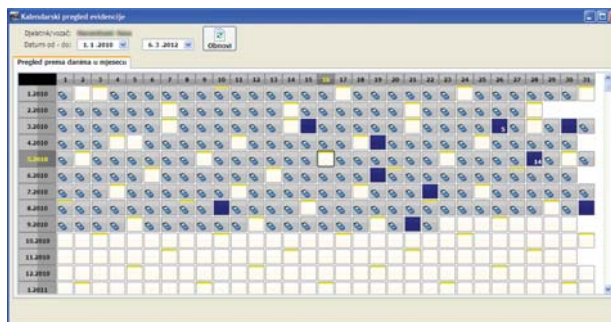


Slika 17. Sučelje programa za obradu listića

Razvijeno rješenje sadrži implementaciju kompletnog postupka opisanog u članku i prikazanog shematski na slici 14. Osim iznesenih algoritama, uključuje i mnoštvo drugih elemenata koji čine kompletan proizvod namijenjen autoprijevoznicima. Očitani podaci se evidentiraju u pouzdanu bazu podataka i povezuju s ostalim evidencijama - podacima s digitalnih tahografa i evidencijom AETR potvrda za slobodne dane, odnosno dane kad mobilni radnici (vozači) ne rade. Posebno je važna funkcionalnost prikaza i vizualizacije evidentiranih podataka. Podaci se prikazuju tablicom i grafički (slika 18.), a analiza je dodatno olakšana mogućnostima kalendarskog i pretraživanog sumarnog pregleda evidencije (slika 19.).



Slika 18. Grafički i tablični prikaz dnevnih aktivnosti



Slika 19. Kalendarski pregled za vozača

Razvoj programskog rješenja potiču korisnici koji nisu bili zadovoljni mogućnostima očitavanja tahografskih listića, a koje su im nudili dostupni i korišteni sustavi. Rješenje problema zahtijevalo je popriličan intelektualni napor, rezultati su zadovoljavajući, a primjena evidencijskog sustava u praksi svakodnevno potvrđuje kvalitetu rješenja.

7. ZAKLJUČAK

U svim državama EU i u državama potpisnicama AETR sporazuma o cestovnom prijevozu robe i putnika autoprijevoznici moraju brinuti o sigurnosti prometa. S tim u vezi propisano je vođenje evidencije o radnim aktivnostima vozača. Kroz praćenje aktivnosti vozača lako se uočavaju prekršaji i propusti koji se nastoje spriječiti.

S obzirom na starost voznog parka, većina autoprijevoznika u Republici Hrvatskoj posjeduje vozila s ugrađenim analognim tahografima. Obaveza vođenja evidencije o radnim aktivnostima vozača uključuje prijenos podataka s papirnatih tahografskih listića na računala. Ovim člankom opisan je prijedlog postupka obrade i očitavanja listića uz minimalna ulaganja u specijaliziranu opremu, koji je, uz dodatke vezane uz evidenciju podataka s digitalnih tahografa, potpuno

implementiran u razvijenom autorskom programskom rješenju. Članak donosi ključne implementirane algoritme, uz detaljno iznesene ideje koje su dovele do njihove primjene.

Iako je razvoj algoritama i pratećeg programskog rješenja opisanog u ovom članku izveden samostalno i neovisno, na temelju jasnih zahtjeva postoji mnogo sličnosti s algoritmima koje su primijenili drugi autori u vrijeme kad su slične potrebe nastale u starijim državama članicama EU ([6], [7]).

Poticaaj razvoju alata za očitavanje tahografskih listića dali su autoprijevoznici koji su se nakon uvođenja pravilnika o vođenju evidencije našli u nezavidnom položaju, s obzirom na to da se dotad nije inzistiralo na evidentiranju u digitalnom obliku. Isto tako, razvoj su dodatno potaknuli i neki korisnici stranih rješenja za evidenciju koji nisu bili zadovoljni pristupom digitalizaciji tahografskih listića. Tako je stvoreno domaće programsko rješenje koje, osim podrške za analogne implementira i podršku za digitalne tahografe i druge propisane evidencije, može zadovoljiti zahtjeve svih naših autoprijevoznika i za koje svakako postoji mjesto na tržištu.

8. LITERATURA

- [1] Zakon o radnom vremenu, obveznim odmorima mobilnih radnika i uređajima za bilježenje u cestovnom prometu, *Narodne novine* 60/08, 124/10
- [2] Pravilnik o prijenosu podataka o radnom vremenu mobilnih radnika i o vođenju evidencije, *Narodne novine* 43/09
- [3] Web – VDO – <http://www.vdo.com>
- [4] Pravilnik o tahografima i ograničavaču brzine, *Narodne novine* 88/08, 48/09
- [5] M. Mikac, V. Mikac "Izvedba programskog rješenja za očitavanje radnih aktivnosti vozača s digitalnih tahografskih kartica" Tehnički glasnik - Časopis Veleučilišta u Varaždinu, 2011 vol 5/2, pp. 21-28
- [6] A. Antonacopoulos, D.P. Kennedy: "Information Extraction from Complex Circular Charts", Proceedings of the 6th International Conference on Document Analysis and Recognition (ICDAR2001), Seattle, USA, September 2001, IEEE Computer Society Press, pp. 784-787
- [7] A. Antonacopoulos, D.P. Kennedy: "An Automated Tachograph Chart Analysis System", Document Analysis Systems V, Springer Lecture Notes in Computer Science, 2002, pp. 544-555

Kontakt:

mr.sc. Matija Mikac, dipl.ing. elektrotehnike
– VELV - Veleučilište u Varaždinu
– e-mail: matija.mikac@velv.hr