

Automatická detekce přírůstku trhliny pomocí digitálního zpracování obrazu

Ondřej Chudý

Ing. Karel Doubrava, Ph. D.

Abstrakt

Obsahem textu je popis algoritmu pro detekci přírůstku trhlín a stanovení velikosti tohoto přírůstku z digitálních snímků povrchu vzorků. Algoritmus je sestaven v prostředí Matlab a využívá především jeho Image Processing Toolbox.

Výsledný algoritmus je schopen vcelku spolehlivé detekce přírůstku trhliny. Vzhledem k různorodosti jednotlivých snímků s trhlínami (širší nebo tenčí trhlina, méně šumu, více šumu) je však vhodné parametry jednotlivých kroků hlídat a vhodně upravit.

Klíčová slova

detekce trhlín, Matlab, konvoluce, vzájemná korelace, otevření obrazu

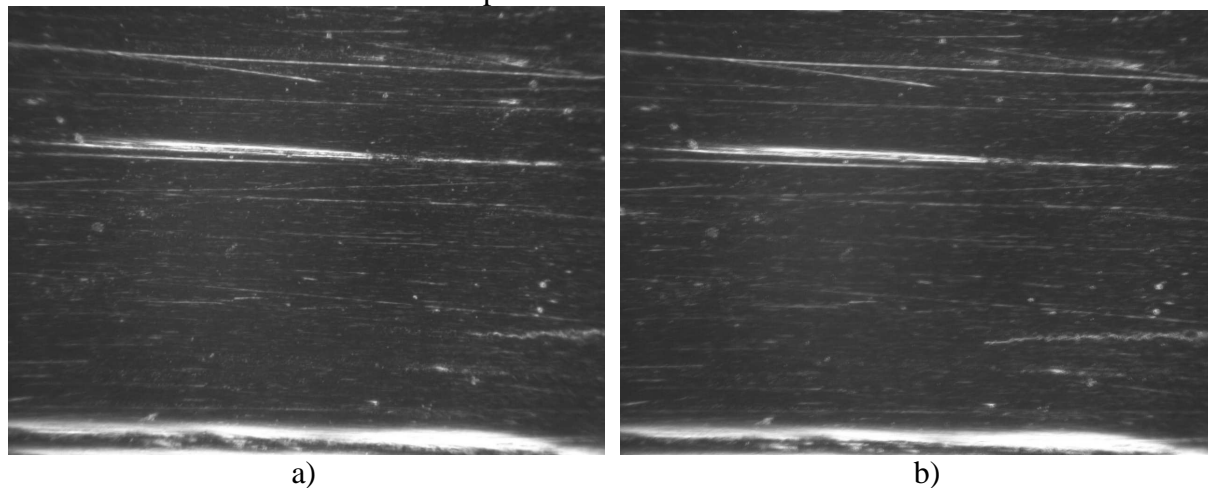
1. Úvod

V textu bude probrán algoritmus určený pro detekci prodloužení trhliny mezi dvěma obrázky. Při popisu bude využito několik metod pro digitální zpracování obrazu. Vzhledem k omezenému rozsahu textu nebude možné popsat tyto metody blíže. Případné informace o použitých metodách, jejich principu a vlastnostech, lze nalézt v uvedené literatuře.

Ke tvorbě následného programu bylo využito prostředí Matlab. Proto budou funkce popisovány tak, jak jsou používány v Matlabu a jak je Matlab zpracovává. Mnoho použitých funkcí je součástí Image Processing Toolboxu.

2. Převod na stupně šedi

Po načtení obrazů do Matlabu je první operací převod do stupňů šedi. Zpracování obrázků ve třech kanálech (RGB) naráz místo jednoho je zbytečně výpočtově náročnější. Informace o přírůstku trhliny tímto převodem neutrpí. V případě leštěných CT vzorků je navíc rozložení barev v RGB obraze velmi blízké stupnici šedé viz Obr. 1.



Obr. 1: Vzorové snímky vybrané pro nástin postupu algoritmu.

V prostředí Matlab bylo využito příkazu „rgb2gray“, který sčítá hodnoty v jednotlivých kanálech násobené váhovými faktory. Konkrétní hodnoty jsou 0,2989 pro červený kanál, 0,587 pro zelený a 0,114 pro modrý.

Výsledný obraz je uložen do jednoho kanálu v 8-bitovém rozsahu pro hodnoty intenzity (matice o stejném rozměru jako má obraz v pixelech^{*}).

Na obr. 1 je dvojice snímků, na kterých budou ilustrovány postupně jednotlivé kroky algoritmu. Jedná se o snímky povrchu CT vzorku s již rostoucí trhlinou. Nachází se v levé dolní části a končí zhruba v 1/3 pravého obrázku.

3. Konvoluční rozmazání a sjednocení intenzit

Po převodu na stupně šedi následuje operace rozmazání (rozostření). Do algoritmu byla přidána kvůli zlepšení výsledků 2-úrovňové filtrace, která později následuje. Každý snímek je snímán s nepatrně odlišnou intenzitou. To může vzniknout nejen díky osvětlení, ale také díky náhodným procesům v samotném snímači intenzity záření – v každém pixelu a velký podíl má i následná diskretizace signálu. Následně při filtraci na 2 úrovně se stává (téměř u každé dvojice snímků), že některé pixely jsou v jednom snímku těsně nad prahovou hodnotou (a tudíž zůstanou zachovány) a ty stejné pixely v druhém snímku jsou těsně pod touto hodnotou (a jsou odstraněny). Tato chyba se potom dál nese algoritmem a někdy může způsobovat potíže, proto je snaha o její eliminaci.

V řešeném příkladu bylo použito jádro:

$$C = \frac{1}{80} \cdot \begin{bmatrix} 2 & 8 & 2 \\ 8 & 40 & 8 \\ 2 & 8 & 2 \end{bmatrix} \quad (1)$$

Zvolené konvoluční jádro je možné v případě potřeby změnit (přepsat na vhodnější pro dané snímky). Za tímto účelem k souboru s algoritmem bude připojen i soubor s konfigurací jednotlivých prvků.

Rozmazáním byla sledována rovněž snaha o potlačení drobných „otřepů“ na hranách trhlin, zejména na jejím čele. Zde totiž tyto „otřepy“ způsobují nepřesnosti při finálním určování velikosti přírůstku, neboť tento je poté počítán včetně těchto „otřepů“.

Pro zlepšení výsledku dvou-úrovňové filtrace byla do algoritmu zařazena ještě operace pro sjednocení intenzit na obou vyšetřovaných snímcích. Pracuje na principu vytvoření váženého průměru hodnoty intenzity na jeden pixel. To je provedeno u obou obrazů a následně je vytvořen poměr těchto hodnot. Tímto poměrem se vynásobí hodnoty všech pixelů jednoho snímku a zaokrouhlí na celá čísla (pracuje se s 8-bitovými obrazy).

Závěrem je třeba přiznat, že efekt těchto dvou operací není 100%. I po jejich aplikaci některé pixely v obrázcích vzájemně „přeskakují“ přes prahovou hodnotu intenzity. Přesto došlo k jistému zlepšení, přesněji úbytku takových pixelů oproti stavu bez těchto operací. Pro svoji výpočtovou nenáročnost byly v algoritmu ponechány.

4. Dvou-úrovňová filtrace a vzájemná korelace

Následující část programu je ve své podstatě příprava pro efektivní vzájemnou korelaci. Ta slouží ke zjištění celkového posunutí obou snímků mezi sebou navzájem (díky mnoha příčinám – zde bude pravděpodobně největší roli hrát zastavení stroje v nestejně poloze). Zjištění a následné odstranění tohoto posuvu je díky způsobu zjišťování přírůstku zcela nezbytné. Nestáčí tedy tuto hodnotu pouze superponovat na výslednou délku trhliny.

* Termíny pixel a bod (rastrového) obrázku je v textu myšlena jedna a stejná věc (obrazový bod). Oba záměnné termíny budou nadále nezávisle použity v textu.

Proces vzájemné korelace je výpočtově velmi náročným procesem. Z hlediska výpočtové náročnosti je to bezkonkurenčně nejnáročnější část celého algoritmu. Proto byly provedeny kroky k její snížení a tím zrychlení celého programu.

Náročnost vzájemné 2D korelace lze vyjádřit vztahem:

$$P_{op} = MN \log_2(MN) \quad (2)$$

kde M a N jsou rozměry obrazu v pixelech. Vzorec platí pro korelaci provedenou pomocí Fourierovy transformace, což odpovídá korelaci v Matlabu pomocí příkazu `xcorr2`. Více v [1] a [2].

Ze vzorce je patrná nejpřímější cesta ke snížení výpočtové náročnosti a to zmenšení rozlišení obrazu. V programu je rozlišení sníženo z původních [1280x960] na [640x480] pixelů, tedy na polovinu.

Dalším krokem pro zrychlení výpočtu je filtrace na 2 úrovně. Po této operaci zbudou v obraze jen 1 a 0 (zde konkrétně hodnoty 0 a 255). Tím se jednak vyloučí případná korelační maxima vzniklá načítáním bodů s různou hodnotou intenzity, za druhé se proces urychlí, protože velká část členů pro násobení jsou nuly. Proces se tím tedy urychlí a zpřesní (jsou zde korelovány posuvy „ostrých“ objektů, nikoliv objektů s pozvolným přechodem do pozadí). Z tohoto důvodu byl nastaven poněkud vyšší práh pro dvou-úrovňovou filtraci, aby zůstalo méně objektů (a bylo více nul). Konkrétní prahová hodnota intenzity byla zvolena 90. Hodnotu lze případně opět jednoduše měnit, praktické testy však ukázaly malý význam této změny. Výsledek této filtrace (a snížení rozlišení – není okem znatelné) je na Obr. 2 (je to filtrovaný Obr. 1 b).

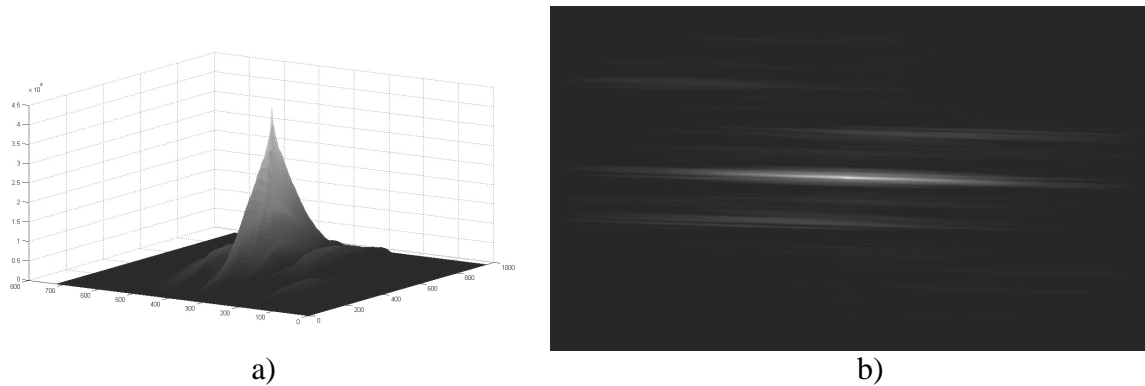
Poté je provedena již samotná vzájemná korelace (příkazem `xcorr2`) a v korelační mapě je z rozdílů souřadnic jejího středu a souřadnic korelačního maxima zjištěn vzájemný posuv obou snímků. Vyjádřen je v pixelech. V této fázi se ovšem jedná o posuv obrazů se sníženým rozlišením. Pro stanovení posuvu původních obrazů je potřeba tyto hodnoty (ve směru x a y) vynásobit poměrem mezi rozlišením původních a korelovaných snímků, zde číslem 2. To samozřejmě zanáší do celého procesu možnou chybu o velikosti jednoho pixelu. Nicméně vzhledem k podstatné úspoře výpočtové náročnosti byla tato nepřesnost shledána za přijatelnou.



Obr. 2.: Snímek 1 b) po dvou-úrovňové filtraci.

Pro další zrychlení výpočtu provedeno oříznutí okrajů u obou snímků. Tím došlo k dalšímu zmenšení počtu vyhodnocovaných bodů. Velikost tohoto oříznutí byla zvolena jako $2 \times 1/8$ obrázku ve směru x a y . Rozměry korelovaných obrazů finálně jsou [361x481].

U zvolené dvojice snímků (na Obr. 1) je korelační mapa na Obr. 3 a). Pro větší názornost byl přidán ještě kolmý pohled „shora“ pro lepší identifikaci konkrétního posuvu. Světlá jsou místa s vyšším korelačním koeficientem. Oboje znázorňuje Obr. 3.



Obr. 3. Korelační mapa pro řešenou dvojici snímků.

Výsledný posuv je zjištěn z rozdílu souřadnice korelačního maxima a geometrického středu korelační mapy (zde má korelační mapa rozměry [721x961]). Na mapě je jasně patrný vliv horizontálních čar (na Obr. 1 a, b), díky nimž lze vysledovat horizontální čáry (a menší horizontální gradient korelačního maxima). Toto je způsobeno velkou vzájemnou korelací (velkou „podobností“) podél těchto objektů (čar). Pokud by snímky obsahovaly pouze objekty přímých trhlin a žádné okolní objekty (šum), např. při chemickém černění vzorků, nastal by problém se správnou identifikací posuvu obou snímků. Proto není zcela vhodné ostatní objekty dokonale odstraňovat.

Hodnoty zjištěného posuvu z korelační mapy na obr. 3 jsou [2, 2] pixelů (směry y a x). Po přepočtu přes poměr rozlišení mezi snímky na Obr. 1 a upravenými snímky pro korelaci je výsledný posuv původní dvojice obrazů roven [4, 4] pixelů.

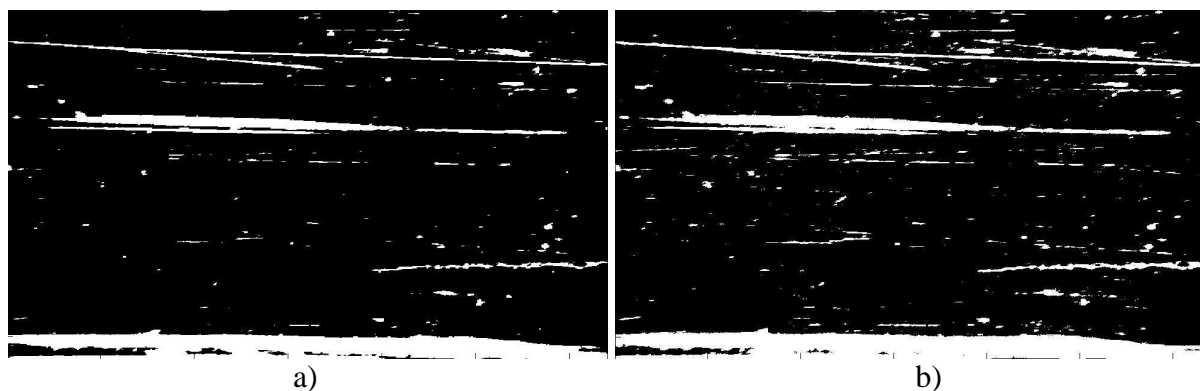
5. Dvou-úrovňová filtrace původních snímků a jejich překrytí

5.1 Dvou-úrovňová filtrace

V další části algoritmu jsou vzaty originální obrázky (tj. „v plném rozlišení“) a opět filtrovány z 8-bitů na 1 bit. Přináší to opět výhody v podobě zrychlení a zjednodušení následujících operací, bez ztráty na jejich přesnosti. Pro následující morfologické operace je tato filtrace navíc nutností (tedy musela by být za každých okolností v průběhu algoritmu provedena).

V této části je již potřeba poněkud „opatrnější“ filtrace původních obrázků. Je potřeba, aby z nich zůstalo zachováno více objektů než pro potřeby korelace. To z důvodu případného neodstranění méně výrazných částí trhliny. Riziko odfiltrování částí trhliny je poněkud zvýšeno konvolučním rozmazáním snímku v úvodní části.

Z tohoto důvodu byla zvolena nižší prahová hodnota intenzity, konkrétně hodnota 80. Pro snímky s výrazně odlišnou intenzitou je vhodné tuto hodnotu přenastavit (bude součástí konfiguračního souboru). Na Obr. 4 je porovnání těchto dvou prahových hodnot (v části a) je Obr. 2). Při soustředění se na trhlínu je v části b) patrné odstranění některých mezer vyskytujících se v části a) a celkový nárůst počtu a velikosti objektů (v části b)). A to i přes relativně malou změnu prahové intenzity.

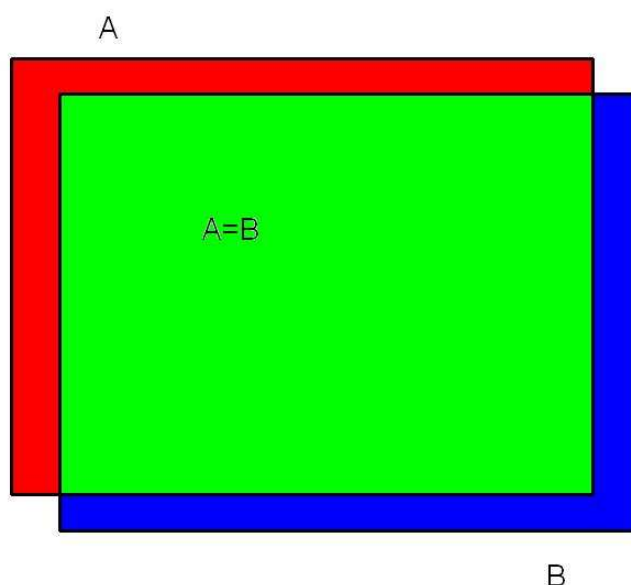


Obr. 4.: 2-úrovňová filtrace Obr. 1 b), 4 a) je pro prahovou hodnotu intenzity 90, 4 b) pro hodnotu 80.

5.2 Překrytí snímků

Po zjištění velikosti vzájemného posuvu následuje posunutí obrazů tak, aby se překrývaly. V programu to bylo řešeno jednoduchým oříznutím příslušných okrajů obou obrazů o velikost posuvu. Dojde při tom k „posunutí indexů pixelů“ a výsledné obrazy si odpovídají.

V prostředí Matlab to bylo řešeno pomocí rozhodovací procedury „if“. Podle směru zjištěného posuvu (kladná či záporná hodnota) je ořezávána příslušná strana obou snímků (v programu značeny jako A a B). Schéma je na Obr. 5.



Obr. 5.: Schématické znázornění potřeby odříznout okraje posunutých obrázků.

Odříznuté části jsou červená (u snímku A) a modrá (ze snímku B). Střední část obrázku je shodná pro oba snímky a zůstává celá zachována.

6. Zjištění přírůstku trhliny

Samotná detekce přírůstku trhliny je založena na rozdílu mezi oběma snímky. Po jejich „srovnání na sebe“ jsou od sebe odečteny hodnoty v jednotlivých pixelech. Po této operaci

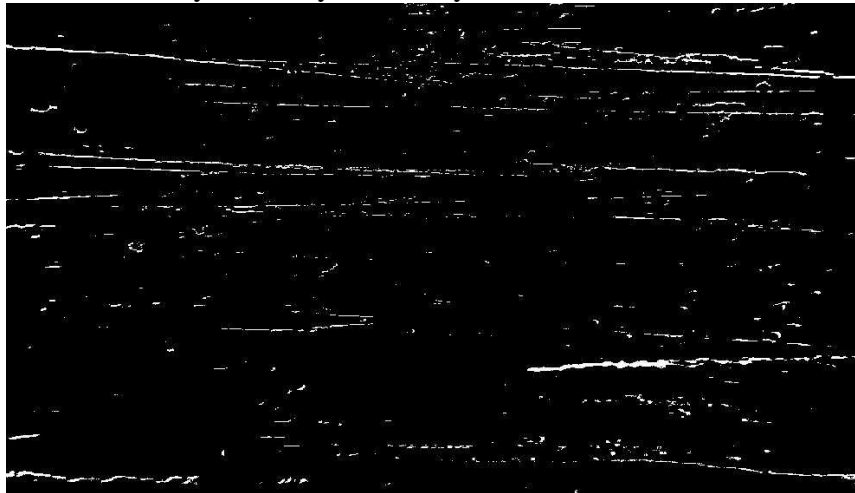
zбудou ve výsledném rozdílovém snímku pouze pixely, jejichž hodnoty jsou vzájemně rozdílné.

Takto jsou odfiltrovány veškeré „šumové“ objekty ve snímcích. Zůstanou pouze objekty, které se nachází na snímku B (Obr. 1 b)) a na snímku A (Obr. 1 a)) chybí. Konkrétně to bude především přírůstek v délce trhliny. Z tohoto se stane nejvýraznější objekt v celém snímku. Kromě tohoto přírůstku zde však jsou další objekty. Odpovídají jednotlivým „rozdílovým“ pixelům, které vznikly díky dvou-úrovňové filtraci. Přesněji díky překmitávání některých pixelů přes prahovou hodnotu intenzity, způsobenou drobnými změnami v intenzitě obou snímků.

Tento postup je rovněž velmi citlivý na jakékoliv změny mezi obrázky. V případě trhlin vznikají problémy díky zvětšování šířky trhliny během dalšího zatěžovacího cyklu (dochází k jejímu většímu rozevření). Což vnáší do rozdílového snímku další objekty.

Problematické je, pokud při snímání obou snímků dojde kromě posunutí polohy vzorku i k jeho rotaci vůči kameře. Vzájemná korelace je schopná dobře zjistit pouze posunutí obrázků jako celku, nikoliv však jejich rotaci (posuvy protějších stran jsou vzájemně opačné). Pro čistou rotaci okolo středu obrázku je poloha korelačního maxima uprostřed korelační mapy, tzn. posuv je nula. I po „srovnání“ obou snímků „na sebe“ zůstanou v rozdílovém snímku části některých objektů v jeho okrajových částech.

Nedokonalosti zmíněné v předchozích odstavcích jsou ilustrovány na následujícím obrázku (Obr. 6). Jde o rozdíl mezi vyšetřovanými snímky z Obr. 1.



Obr. 6.: Rozdíl vzorových Obr. 1 a), b).

Snímky byly podrobeny korekci jejich posunutí, ale díky rotaci mezi těmito snímky zůstaly v jejich rozdíl zbytky některých objektů. Samotný přírůstek trhliny je nejvýraznější bílá čára v dolní pravé části snímku.

Byly provedeny pokusy i o korekci vzájemné rotace snímků pomocí Radonové transformace. Pokusy však nebyly úspěšné. Pravděpodobná příčina je spatřována v nevýrazné směrové orientaci objektů v obrázcích. Maximum hodnoty úhlu nebylo výrazné a měla na něj vliv změna intenzity snímku, či jejich posunutí.

Radonová transformace při praktických pokusech detekovala jisté natočení, které ale neodpovídalo zpravidla skutečnému natočení snímků. V některých případech došlo k detekci vzájemného natočení i u snímků, které natočené nebyly.

V celkovém měřítku došlo díky Radonové transformaci ke zhoršení výsledku, tzn. ve výsledném „rozdílovém“ snímku bylo více šumu (dalších objektů mimo přírůstek trhliny), než když byla korekce natočení vynechána.

7. Transformace otevření obrazu

Pro odstranění drobných objektů, které zbyly po odečtení vyšetřovaných snímků, byla jako nejvhodnější vybrána transformace otevření obrazu. Ta je vhodná na odstranění drobných objektů z binárních obrázků.

Zde je již potřeba pečlivěji sledovat celkový průběh této transformace, aby nedošlo k případnému zkrácení, či odstranění některé části trhliny. V případě takového výsledku je třeba upravit strukturální elementy, v některých případech bylo vhodné jednu část této transformace vynechat. Proto jsou jednotlivé mezikroky v této transformaci vykreslovány.

Transformace otevření obrazu byla aplikována postupně ve dvou fázích, v každé fázi s jiným strukturálním elementem. Námi vyšetřované snímky vykazují (díky rotaci) značné množství nadbytečných objektů (šumu), musely být použity relativně „hrubé“ strukturální elementy na jejich eliminaci (alespoň částečnou).

Pro první otevření obrazu byl použit strukturální element:

$$G_1 = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \quad (3)$$

Výsledek po transformaci je na Obr. 1. Je vhodné provést srovnání s Obr. 6, který znázorňuje stav před touto transformací.



Obr. 7.: Obr. 6 po první aplikaci transformace otevření obrazu.

Pro druhou fázi by použit strukturální element G_2 .

$$G_2 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad (4)$$

Finální výsledek je na dalším obrázku (Obr. 8). Srovnání s předchozími obrázky (Obr. 6 a Obr. 7) vypovídá o prokazatelném zlepšení (odfiltrování šumu) a tím i smyslu (a robustnosti) morfologických transformací.



Obr. 8.: Finální obrázek s přírůstkem trhliny. Obr. 7 po druhém otevření obrazu.

Při kvalitním „rozdílovém“ snímku (tj. obsahuje především přírůstek trhliny a málo šumu) je vhodné některou z transformací vynechat. Nejjednodušší způsob je změna příslušného strukturního elementu na $G = [1]$. Samozřejmě lze i celou operaci ve zdrojovém kódu vymazat (přeměnit na komentář).

V případě tenké trhliny je vhodné strukturní elementy zmenšit (především ve vertikálním rozměru). Pokud budou použity elementy podobně hrubé jako (3) a (4), může dojít k odfiltrování (vymizení) podstatné části trhliny. A to nejen na okrajích, ale i „uprostřed“ přírůstku. To vede ke komplikacím v další části algoritmu, která se věnuje již samotné detekci a kvantifikaci přírůstku trhliny.

Limitem filtrování pomocí operací otevření obrazu je případ, kdy máme tenkou trhlínu a výrazné nadbytečné objekty (např. díky rotaci snímku). V takovém případě je bohužel pravděpodobné selhání celého algoritmu.

8. Vyhledání a kvantifikace přírůstku trhliny

8.1 Vyhledání přírůstku trhliny

Pokud bychom po předchozím zpracování obrazu dostali obrázek, který bude obsahovat pouze přírůstek trhliny, byla by detekce a kvantifikace přírůstku velmi snadnou záležitostí. Stačil by výpis všech bodů s hodnotou 1 (nebo 255) a poté vybrat souřadnice nejzazších bodů (vpravo, vlevo, nahoře, dole). Z toho se již snadno získá délka (horizontální, vertikální, Euklidovská). Díky předchozímu zpracování obrazu není dosažení takového výsledku výjimečné. Nicméně i z pohledu na vzorové, řešené snímky je patrné, že též není výjimečné nedosažení tohoto „ideálního“ stavu.

Z těchto důvodů byl dále sestaven alespoň jednoduchý algoritmus na prohledání výsledného snímku (Obr. 8).

Základem algoritmu je rozdělení celého obrázku (Obr. 8) na menší podoblasti. V konkrétním případě je to rozdělení obrázku s rozměry [956, 1276] pixelů (tyto rozměry mají snímky po oříznutí okrajů po korelaci) na oblasti s rozměry [16, 22] pixelů. Podoblasti jsou srovnány do tabulky (matice, rozměr [60, 59]), kde na každém poli je jedna podoblast.

Základem následujícího kroku je předpoklad, že přírůstek trhliny je nejvýraznější objekt v celém obraze („zpracovaném“ Obr. 8), nebo že má alespoň největší hustotu bodů. S tímto předpokladem jsou prohledány všechny podoblasti a zaznamenány počty nenulových bodů (s

hodnotou 1 či 255) uvnitř každé podoblasti. Poté je vybrána podoblast s největším počtem nenulových bodů. O té je předpokládáno, že je to jedna z podoblastí, v nichž se nachází hledaný přírůstek trhliny. Tato podoblast je dále „startovací“ oblastí pro další část prohledávání. Po nalezení části trhliny a stanovení „startovní“ podoblasti je prohledáváno její okolí do všech směrů (s výjimkou směru zpět). Důvodem je snaha o jistou robustnost, univerzálnost algoritmu (aby byl schopen případně sledovat trhliny jdoucí svisle, případně i v obecném směru). Navíc též horizontální trhliny se nemusí šířit po přímcích, ale (např. díky nehomogenitě materiálu) jsou schopné svůj směr lehce měnit, či „uskočit“ ve směru kolmém na směr šíření. Předpokládá se, že růst trhliny nebude probíhat směrem zpět, nejhůře směrem kolmo na předpokládaný směr šíření.

Algoritmus je postavený tak, aby v daném směru nejprve prověřil nejbližší (sousední) oblast. Rozhodujícím kritériem je přítomnost nenulových bodů ve vyšetřované podoblasti. Pokud zde jsou, je tato podoblast brána jako další oblast s trhlinou, algoritmus si uloží souřadnice původní oblasti a přesune se do nové. Odtud se celý postup opakuje.

Pokud nenulové body nejsou přítomny, jsou vyšetřovány nejbližší okolní podoblasti ve směru šikmém a kolmém na směr šíření. Příklad celého postupu je na Obr. 9 pro směr podle záporné osy x . Čísla odpovídají pořadí, v jakém jsou jednotlivé podoblasti (zde znázorněna jako políčka) prohledávány.

	3	4	
	1	Startovací oblast	
	2	5	

Obr. 9.: Ilustrace hledání další podoblast s trhlinou.

Takto algoritmus pokračuje dokud nenarazí na konec trhliny v daném směru. Poté, opět z původní startovací pozice, startuje ve směru dalším.

Na konci trhliny v některých případech hrozí „zacyklení“ algoritmu, např. na konci trhliny mohou najít nenulové body až v oblasti „5“ (trhlina leží na hranici oblastí), přesunu do ní aktuální startovací oblast a znovu prohledám okolí, pak naleznou nenulové body vždy v oblasti 4, což je předchozí startovací pozice apod. Proto bylo nutné kontrolovat, jestli se minulá startovací oblast (uložená jako předposlední v paměti) neshoduje s budoucí, zatím vyšetřovanou. V takovém případě je postup ukončen a přejde na prohledávání v dalším směru (z původní startovací oblasti).

Rovněž je nutné kontrolovat, aby algoritmus při prohledávání nezašel za okraje matice podoblastí. Toto bylo ošetřeno jednoduchými pevnými mezemi pro hodnoty souřadnic $(0, x)$, $(0, y)$.

Pro zvýšení odolnosti algoritmu vůči drobným objektům v těsném okolí trhliny, které tam i po dvojitě provedení operace otevření obrazu mohou zůstat, je nastaven minimální počet nenulových bodů na 9. Hodnota 9 odpovídá počtu jedniček ve strukturním elementu G_2 . Příčinou je situace, kdy po první části operace otevření obrazu (eroze) z objektů v okolí trhlín (šumu) zůstane pouze „centrální“ bod příslušného objektu. Následující operace dilatace kolem osa-

mocného bodu vytvoří objekt se shodnou velikostí jako má použitý strukturní element. Díky minimálnímu počtu bodů algoritmus takový objekt vynechá. Pokud vyšetřovaná podoblast obsahuje 9 a méně nenulových bodů, není považována za další podoblast s trhlinou.

8.2 Kvantifikace přírůstku trhliny

Po předchozí části vznikne tabulka (matice) souřadnic oblastí, v nichž byla nalezena trhlina. Z tabulky jsou vybrány maximální a minimální hodnoty souřadnic a z původního snímku je vybrána opsaná obecně obdélníková oblast, která bude přesně zahrnovat všechny nalezené podoblasti. Jedná se v podstatě o obdélníkový výřez okolo trhliny z původního snímku. Výsledek námi řešeného případu je na Obr. 10. Jedná se o výřez z Obr. 8.



Obr. 10.: Výřez přírůstku trhliny z Obr. 8.

Pro možnost kontroly je tento výřez vždy vykreslován. Může nastat situace, že oblast obsahuje i nějaký bod mimo trhlinu vzniklý díky šumu. Pokud se tak stane, je vhodné zmenšit velikost jednotlivých podoblastí, nebo zvětšit minimální počet nenulových bodů pro každou oblast (obojí lze snadno nastavit).

Další možnou situací je vymizení části trhliny. Dojít k tomu může dvěma způsoby. Prvním způsobem je vymazání části trhliny během transformací otevření obrazu. Jedno z možných řešení je zmíněno výše (zmenšení strukturního elementu). Druhou příčinou je „projetí“ trhliny nějakým objektem (zpravidla rýha, oděrka, nebo nečistota na povrchu vzorku), který je přítomen v obraze i před šířením trhliny. Po zpracování (konkrétně při operaci vzájemného odečtení snímků) vznikne takto v trhlině „proluka“.

Podle aktuální velikosti vzniklých „proluk“ hrozí, že algoritmus na jedné straně této mezery zastaví a nedostane se přes ni. Řešením je odpovídající zvětšení podoblastí tak, aby byly schopny tuto mezeru překlenout.

Problematické je, pokud trhlina v takovém objektu (rýha, oděrka apod.) skončí. Potom již algoritmus není schopen určit skutečný konec trhliny. Takový úkol je však problematický i pro lidské oko a mozek.

Samotné stanovení délky přírůstku je provedeno pomocí rozdílu nejvzdálenějších bodů ve výřezu (souřadnice konců přírůstku trhliny). Pomocí znalosti velikosti jednoho pixelu ve skutečnosti (přesněji rozměru celého snímku) je vypočtena horizontální, vertikální a Euklidovská délka trhliny. Dále jsou stanoveny souřadnice konců trhliny v celkovém snímku – přičtením souřadnic výřezu k souřadnicím konců trhlín uvnitř výřezu.

Za účelem případného posunutí snímací kamery je stanovena a vypsána velikost vzdálenosti konce trhliny (zde levý konec přírůstku – šíření trhliny bylo zprava doleva) od středu snímku. To umožňuje polohování kamery tak, aby další přírůstek trhliny byl pokud možno uprostřed snímku, tedy v místě největší ostroty kamery. Posuv je zaokrouhlen, v našem případě na setiny milimetru (zde $[-0,22, -0,14]$).

9. Závěr

Algoritmus se při testování ukázal jako vcelku robustní a univerzální. Má i některé limity (zmiňované v textu), například není příliš odolný vůči vzájemné rotaci snímků, dále má problémy pokud trhlina projede objektem již v obraze přítomným atd.

Z tohoto důvodu jsou důležité mezikroky vykreslovány, aby bylo možné ihned přenastavit příslušné parametry pro jednotlivé kroky. Zde například pokud je použit příliš hrubý strukturální element pro otevření obrazu, který způsobí vymazání nejen šumu, ale i části trhliny.

Problémy někdy způsobuje i přílišná citlivost algoritmu na veškeré změny mezi snímky. Díky této citlivosti může být rozměr trhliny zvětšen o ořepky na čele trhliny (které ale doposud trhlinou nejsou), případně vznikají další výrazné „šumové“ objekty při sekundárním rozšiřování předchozí části trhliny. V tomto případě je třeba zmenšit velikost podoblastí pro prohledávací část algoritmu.

Obecně ale stanovuje přírůstek trhliny s odchylkou $\pm 10 \mu\text{m}$.

Použitá literatura

- [1] Jähde Bernd: Digital Image Processing, Springer – Verlag Berlin, Berlin 2002, (585 stran)
- [2] Prof. Ing. Hlaváč Václav, CSc., Doc. Ing. Miloš Sedláček, CSc.: Zpracování signálů a obrazů, Vydavatelství ČVUT v Praze, Praha 2007, (255 stran)
- [3] Bastiaans Rob J.M.: Cross-correlation PIV, Dostupné z URL: <http://alexandria.tue.nl/repository/books/531139.pdf> (citováno 4. 10. 2011)
- [4] The Math Works Inc., Matlab R2011b Documentation, Radon Transform, Dostupné z: <http://www.mathworks.com/help/toolbox/images/f21-25938.html> (citováno 20. 1. 2012)
- [5] Collet Chritophe, Chanussot Jocelyn, Chehdi Kacem: Multivariate Image Processing, ISTLE, London 2010 (459 stran)