

Задача

Постановка

Ден 34 от извънредното положение. След края на последния епизод на третия сериал, който гледам тази седмица, всичко започва да губи своя смисъл. Може би четвърти сериал ще оправи нещата? Какво имаме тук... този готварски курс изглежда добре. Тази поредица от мисли плавно прелива в осъзнаването, че всъщност съм доста гладен. Пътешествието до хладилника не дава положителен резултат, и след още 10-минутно размишления и второ безрезултатно пътешествие до хладилника, дори със занижени стандарти, става ясно, че ще има нужда от по-продължителна експедиция. С отчаяна въздишка надявам якето си и излизам по анцунг в мъгливия пролетен въздух.

Докато вървя по тротоара на тъмната уличка, която стига до моя блок, виждам фигура на човек с противогаз, облегнал се на оградата. Докато го наближавам, си мисля: ех, голям страх с това извънредно положение - маски, шалове... но чак противогаз? Той обаче се обръща към мен точно в момента, в който го подминавам - може би трябваше да ме е страх от произволен човек с противогаз и да мина по другия тротоар?

- Глабяркасианаяиф!
 - Извинете?
 - Здравейте, бихте ли искали да превъртите времето назад и да подредите живота си по друг начин? — *монотонен глас с леко грешни ударения*
 - Какъв е този въпрос? — загъръщам се с якето и го подминавам
- Тягостното настроение от по-рано се завръща с пълна сила. Какъв злонамерен план има този човек и защо задава такива странини въпроси? Всъщност, той е просто герой в някаква тъпа задача, защо би бил злонамерен?

Връщам се при господина (госпожата?) с противогаза и заявявам:

- Всъщност, да. Искам да се въртате.
- Той (тя?) поставя ръка на рамото ми, преди да съм се изказал докрай.

- Направи ли си проекта за утре? — *казва майка ми*
- д–да. — *казвам аз след дълга пауза*
- Всичко наред ли е? — *пита ме тя загрижено*
- Да, всичко е наред.
- Лека нощ! — *казва тя и гаси лампата*

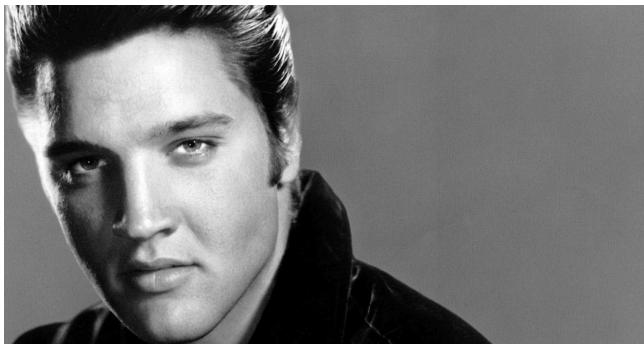
Аз съм в леглото, което имах като малък. Ставам тихо, включвам лампата и поглеждам в огледалото. Ситуацията е толкова невероятна, че я приемам без никаква емоция. Аз съм отново петокласник. Пускам компютъра. Няма парола. Колко непредпазлив съм бил! „Проектът за утре“ е за проект по музика, за който всеки трябва да направи албум от снимки на някой изпълнител. Аз съм изтеглил един торент със снимки на Елвис, но естествено, както много добре помня, това че е готов е пълна лъжа: снимките са с лоша композиция и различни размери. АзКатоПетокласник би бил напълно задоволен с това. Мразя АзКатоПетокласник. Има цялото време на света, а седи по цял ден и си бърка в носа. Поглеждам часовника - 7 часа до закуска. Отварям браузър и въвеждам „debian.org“ в полето за адрес.

Този път ще направя всичко като хората. С много старание и без измама.

План

Имаме директория, която съдържа снимки. Тези снимки са с най-различни размери. Дадено ни е някакво съотношение, в което трябва да изрежем всички снимки (за примерите ще използваме съотношение 3:2, но скриптовете, които пишем, трябва да работят с произволно). Най-простият начин да изрежем снимка е да отрежем прозореца в центъра, но това не винаги дава най-добрая резултат (например, следната картичка е по-красиво да я изрежем с прозорец, който е вляво, отколкото прозорец, който е в центъра):

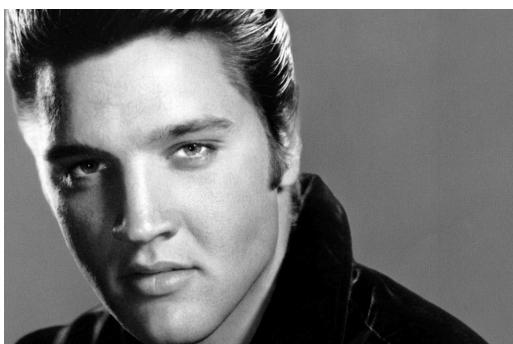
1. Оригиналното изображение



2. Прозорец в центъра



3. Прозорец вляво



Затова, за всяка снимка ще определим „интересна точка”, и ще се опитаме да генерираме прозорец, който е максимално голям, центърът му е максимално близо до интересната точка, и съотношението му е правилно.

Помощни материали

Днес ще обработваме изображения. За целта, на сървъра astero.openfmi.net са предоставени следните инструменти:

скрипт *pw*

С него можем да преглеждаме изображения. Подаваме изображения като аргументи и скриптът връща на *stdout* линк, който можем да използваме, за да ги видим визуално:

```
$ pw pika.png  
https://astero.openfmi.net/preview/f563c5sbz1
```

Можем да подадем повече от един аргумент. Също, можем да *hover*-ваме с мишката над изображението, което ни показва координатите на пиксела под мишката. Тази възможност ще ни е удобна по-нататък.

imagemagick toolkit

Той се състои от следните компоненти (и други, които няма да ползваме):

- команда *identify*, която връща информация за подадено изображение
- команда *convert*, която извършва манипулации върху подадено изображение и записва резултата в нов файл

Въсъщност, на нас ще ни трябват само две конструкции, които даваме тук, за да си спестим четене на документацията:

- *identify foo.png*, за да видим размера на изображението *foo.png*
- *convert foo.png -crop 200x300+10+50 +repage bar.png*, за да изрежем правоъгълник с размер 200x300 с горен ляв ъгъл на координати (10, 50) от изображението *foo.png* и да запишем резултата в ново изображение *bar.png*

Документация: <http://www.imagemagick.org/Usage/>

команда *facedetect*

С нея можем да откриваме човешки лица в изображения.

Например:

```
$ facedetect elvis.jpg  
485 105 95 95  
816 138 84 84
```

facedetect извежда информация за всяко открито лице на отделен ред. Без допълнителни параметри, извежда 4 числа, които са координати на горен-лев ъгъл и размер на правоъгълника, описващ лицето.

На нас ще ни свърши работа по-простиия вариант, *facedetect -c*, който извежда само координатите на центровете на откритите лица:

```
$ facedetect -c elvis.jpg  
532 152  
858 180
```



Процес на работа

На сървъра има директория /tmp/elvises, в която има няколко примерни входни снимки. Може да ги използвате, за да тествате скриптовете си.

Първа част

Напишете скрипт crop.sh, който при подаден първи аргумент ipoint, се извиква по следния начин:

```
$ crop.sh ipoint <a> <b> <x> <y> <input image> <output image>
```

Където:

- <a> и дефинират съотношение на търсения размер
- <x> и <y> дефинират координати на интересна точка
- <input image> и <output image> са съответно входен и изходен файл

Например:

```
$ crop.sh ipoint 3 2 150 321 foo.jpg foo_cropped.jpg
```

Вашият скрипт трябва да изреже изображението във входния файл и да запише резултата в изходния файл, така че:

- размерът на изходния файл да има съотношение a:b, ако a > b и sx > sy или да има съотношение b:a, ако a > b и sy > sx, където входното изображение има размер sx : sy. С други думи, портретните снимки остават портретни, а пейзажните остават пейзажни. Ако е подадено b > a, може да ги размените или да изведете съобщение за грешка, по ваш избор.
- да няма начин да се изреже изображението, така че да получим резултат с по-голям размер, или да получим резултат, в който интересната точка е по-близо до центъра.

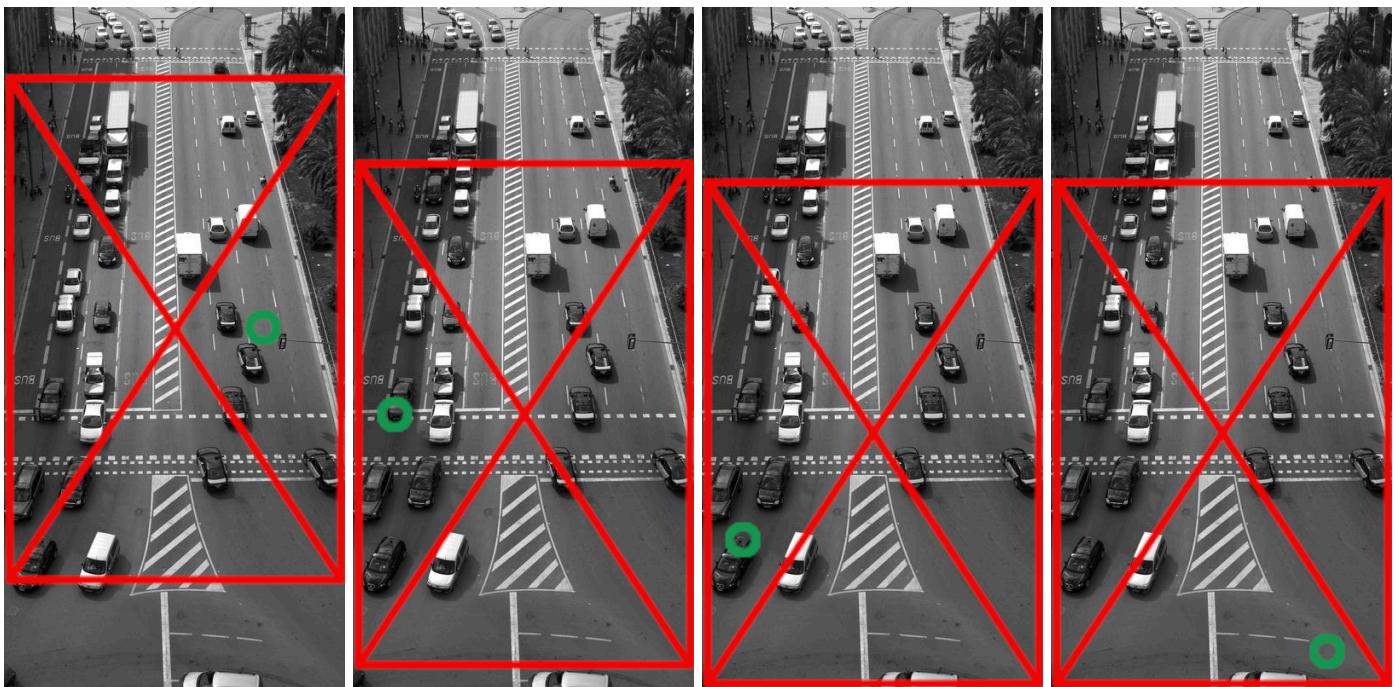
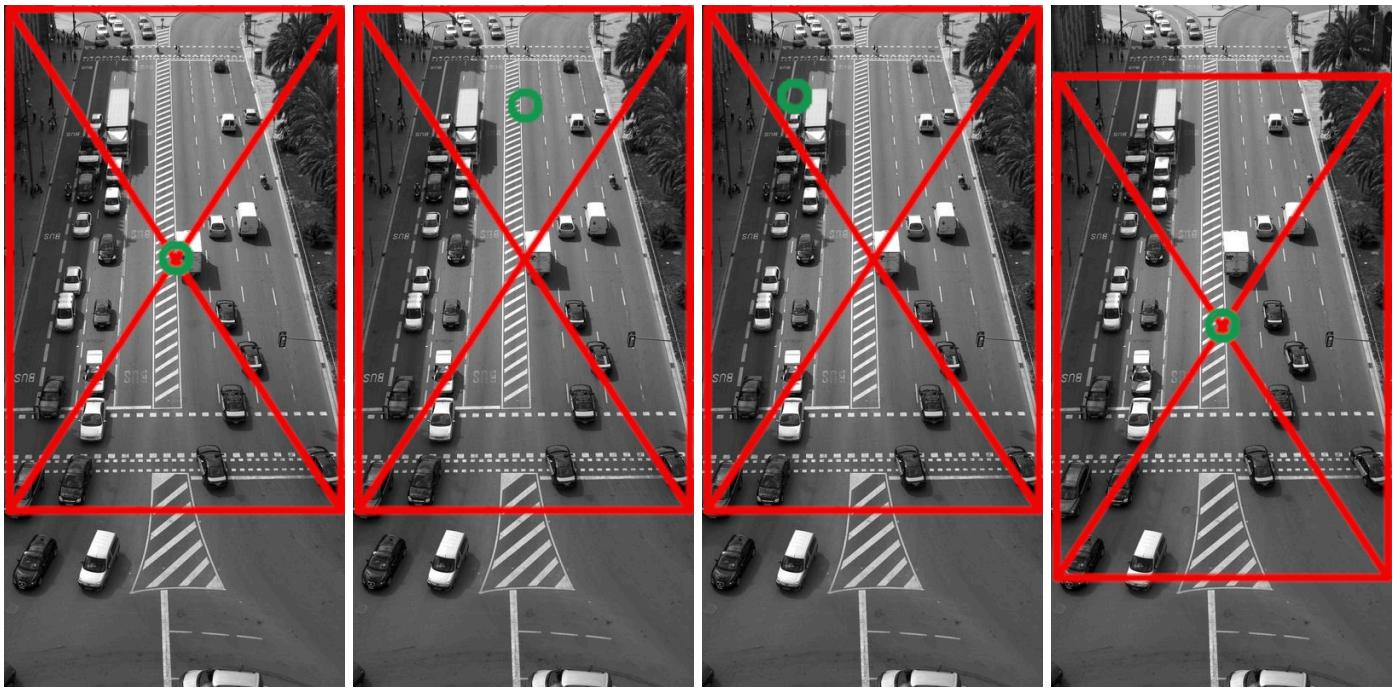
Тъй като това описание звучи прекалено сложно, вместо да го четем няколко пъти, за да го разберем, ще е по-лесно да погледнем следните примери:

Примери

Тези примери показват всички възможни случаи, които трябва да бъдат обработени. На всеки от примерите имаме входното изображение, като върху него с зелено кръгче (○) е показана интересната точка, а с червена рамка (☒) е показана областта, която вашият скрипт трябва да изреже.

Във всеки от тези примери $a:b = 3:2$.





Втора част

Допълнете скрипта `crop.sh`, така че при подаден първи аргумент `faces`, той се извиква по следния начин:

```
$ crop.sh faces <a> <b> <input image> <output image>
```

и работи по същия начин както варианта с `ipoint`, само че вместо да приема интересната точка като аргумент, избира за интересна точка медианата на всички човешки лица в изображението. Ако няма човешки лица, скриптът избира центъра на изображението за интересна точка.

Трета част

Напишете скрипт `crop_all.sh`, който се извиква по следния начин:

```
$ crop_all.sh <a> <b> <in_dir> <out_dir>
```

Където `<a>` и `` пак дефинират съотношение, `<in_dir>` е директория, в която има много на брой изображения, а `<out_dir>` е директория, в която скриптът трябва да записва резултатните изображения.

Скриптът ви трябва да минава през всяко изображение, и да показва на потребителя избор между три евристики, така че той лесно да може да види резултатните изображения и да си избере:

- Център на изображението
- Медианата на човешките лица
- Ръчно въвеждане на координати на интересна точка