

První náhled

Nejdřív náhlédneme na rozsah složitosti mediánového filtru. Ze zadání nám bylo prozrazeno, že naivní řešení vychází na komplexitu $\Theta(N^2 \cdot K^2)$. Můžeme dál snadno nahlédnout, že úloha nejde řešit lépe než $\mathcal{O}(N^2 \cdot K)$.

Důkaz: Ze zadání víme, že máme N^2 různých pixelů. Každý z těchto pixelů má nějaké obklopující pixely, jinak řečeno, vstup pro mediánový filtr. Každý pixel může mít unikátní vstup pro mediánový filtr. Tedy, nevyhneme se procházení všech pixelů a hledání mediánu pro každého pixelu zvlášť.

Sice určení mediánu pro první pixel může trvat K^2 kroků, abychom načetli vedlejší pixely, ale poté co toto máme načtené, můžeme vstup pro vedlejší pixel načíst tak, že odstraníme jednu stranu čtverce a poté přidáme stranu čtverce na té straně, kudy se pousouváme. Vlastně jenom posouváme ten čtverec pro mediánový filtr o 1 pixel v *nějakém* směru. Tedy po prvním pixelu musíme provést vždy *minimálně* K kroků (v tomto případě nepočítám pixely na kraji, protože nijak nezrychlují algoritmus pro ostatní pixely).

Také bych ještě zmínil, že protože potřebujeme původní údaje na provádění mediánového filtru, vypočítané hodnoty si musíme uložit jinde, tedy budeme mít prostorovou složitost $\mathcal{O}(N^2)$. I tak budu zmiňovat prostorovou složitost u řešení (bez zmiňování N^2), protože si myslím, že to může být i tak vhodný faktor pro rozhodování, co použít.

První řešení

Teď, když víme, že se pohybujeme mezi $\mathcal{O}(N^2 \cdot K^2)$ a $\mathcal{O}(N^2 \cdot K)$, je zřejmé, že hledáme lepší způsoby hledání mediánu. První taková optimalizace je dost jednoduchá. Hodnoty ze kterých budeme počítat medián si uložíme v dictionary u každého z nich si budeme pamatovat jeho počet ve čtverci. Toto nám umožní jednoduché vkládání a odstranění v konstantním čase, tedy opravdu strávíme jen K kroků při přesunu na další pixel. Limitujícím faktorem (pravděpodobně) je výpočet mediánu.

Abychom ho vypočítali, musíme projít všechny hodnoty a zjistit, který z nich obsahuje prostřední člen. Toto je lineárně závislé na počet unikátních hodnot ve čtverci, což tedy může být kdekoli od 1 do K^2 . Tedy, tento algoritmus má nejhorší složitost $\mathcal{O}(N^2 \cdot K^2)$ a případně nejlepší $\Omega(N^2 \cdot K)$ (protože stále musíme posouvat čtverec, situace kdy počet unikátních hodnot je menší než K nás nezajímá).

Prostorová složitost tohoto algoritmu je také závislá na počet unikátních hodnot, to tedy vychází jako $\mathcal{O}(K^2)$.

Druhé řešení

Sice je předchozí řešení rychlejší, ale její rychlost je velice variabilní a spíš bude větší než $N^2 \cdot K$, třeba okolo nějakých $N^2 \cdot K\sqrt{K}$. Proto jsem vymyslel i další řešení.

Tentokrát místo dictionary, budeme mít intervalový strom. Tento intervalový strom bude mít v sobě uložený, kolik celkem hodnot jsou v levé a v pravé půlce a takhle můžeme poslat dotaz pro hodnotu na místě $(K^2)/2$.

Nejdřív projdeme všech N^2 pixelů, najdeme největší hodnotu pixelu a označíme ho M . Základem intervalého stromu bude array o délce M a každý index si ukládá kolik pixelů (ve čtverci) mají

stejnou hodnotu jako index. Poté postavíme nad tímto arrayem binární intervalový strom, který si pamatuje, kolik pixelů jsou v levém a pravém podstromu.

Když jsou hodnoty uloženy a chceme najít medián, tak můžeme konstantním dotazem na kořen stromu zjistit, kolikátou hodnotu hledáme a poté hledání hodnoty samotné můžeme najít v $\log M$ čase. Hledání mediánu pro každý pixel tedy potrvá $N^2 \cdot \log M$ času, ale bohužel nás limituje vkládání a odebírání hodnot do intervalého stromu. Vlastně, vkládání a odebírání samotné je konstantní, ale aktualizace hodnot větví intervalého stromu vyžaduje $\log M$ času, což na celé posouvání čtverce vychází na $\mathcal{O}(K \cdot \log M)$.

Celková složitost tohoto řešení je $\Theta(N^2 \cdot K \cdot \log M)$ a prostorová složitost $\mathcal{O}(M)$.

Které řešení je lepší

Když budeme uvažovat jenom nejhorší rychlost, tak hledáme jenom kdy $\log M < K$. Víme, že K je řádově větší než nějaká malá konstanta, takže pro úplně nejnižší odhad můžeme říct, že $K = 10$. V takovém případě, M může být až $2^{10} = 1024$, než bude druhé řešení vždy horší. Vůbec, v případě, kdy $\log M$ se blíží ke K , tak první řešení bude spíš rychlejší, kvůli jeho velké variabilitě. Předpoklad $K = 10$ je sice nejnižší podle zadání, ale pro fotky na uživatelské úrovni, příliš velký mediánový filtr jen fotku rozmaže.

Sice nevím, jak velké jsou fotky z dalekohledů, ale je zaručeno, že jsou větší než typické fotky a v takovém případě můžeme předpokládat větší K .

Už jen předpoklad $K = 15$ nás nechá umožnit až $M = 2^{15} = 32768$ a samozřejmě další malé zvětšení K nám dá mnohem větší limit pro M , než druhé řešení bude vždy horší než první.

Proč nikdy nedosáhneme $\mathcal{O}(N^2 \cdot K)$

Jak už bylo řečeno, už jen procházení všech pixelů trvá $N^2 \cdot K$ času. Abychom tuto složitost dodrželi, museli bychom dokázat hledání mediánu až K^2 hodnot v K času, tedy pro nějakých X hodnot najdeme medián v \sqrt{X} čase, což je logicky jen tak nemožné. Můžeme najít medián rychleji pomocí jiných způsobů, jako ten popsáný v druhém řešení, ale takové metody nám neumožní zanechat tu základní složitost $N^2 \cdot K$, kvůli přídatných operací při posunu čtverce (například, aby se zachoval tříděný seznam hodnot).