

Semestrální projekt předmětu Výpočetní fotograbie – závěrečná zpráva  
**Tone Mapping: implementace algoritmu Khan20**

*Milan Tichavský, xticha09@fit.vut.cz*

# 1 Úvod

Zařízení běžně používaná k zobrazování digitálního obsahu, jako jsou monitory, televizory a tiskárny, nejsou schopna zobrazit celý dynamický rozsah světla přítomného v reálném světě. Z tohoto důvodu existují metody pro převod obrazu s vysokým dynamickým rozsahem (HDR) na standardní dynamický rozsah (SDR), aby bylo možné jej správně vizualizovat na dostupných zařízeních. Tento proces se nazývá "tone mapping".

Cílem této práce je implementace a analýza jednoho z moderních tone-mapping algoritmů, konkrétně metody Khan20 [1], jako pluginu do Tone Mapping Studio (TMS) [2].

## 2 Teoretické základy

Pro pochopení tone-mappingu je nutné zmínit vlastnosti lidského vidění. Lidské oko dokáže adaptivně vnímat velký rozsah jasových hodnot díky nelineárnímu vnímání jasu a kontrastu. Algoritmy tone-mappingu se často inspirují těmito vlastnostmi a snaží se přizpůsobit zobrazení tak, aby výsledný obraz působil přirozeně.

## 3 Popis algoritmu Khan20

Algoritmus Khan20 je globální tone-mapping operator (TMO), to znamená že pro transformaci je využita monotónní neklesající mapovací funkce. Algoritmus je založený na transformaci obrazu pomocí funkce *Perceptual Quantizer (PQ)* a histogramu jasu. Schéma algoritmu je vidět na obrázku 1.

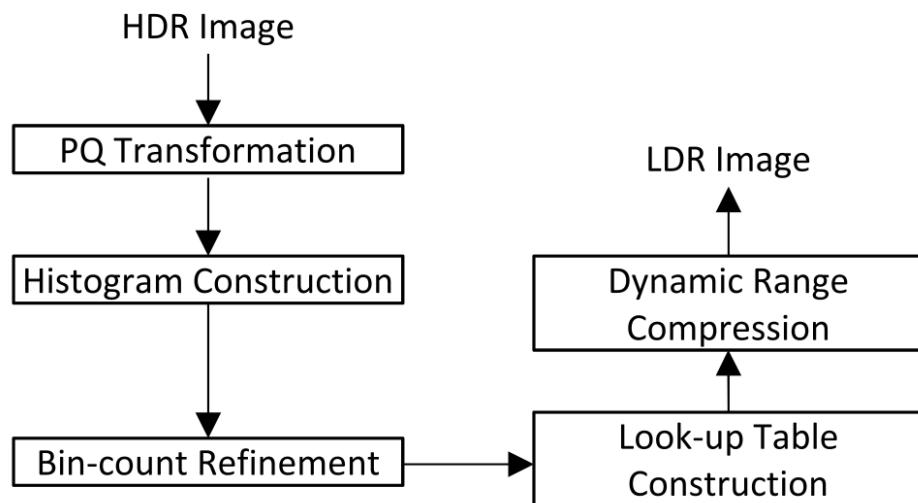
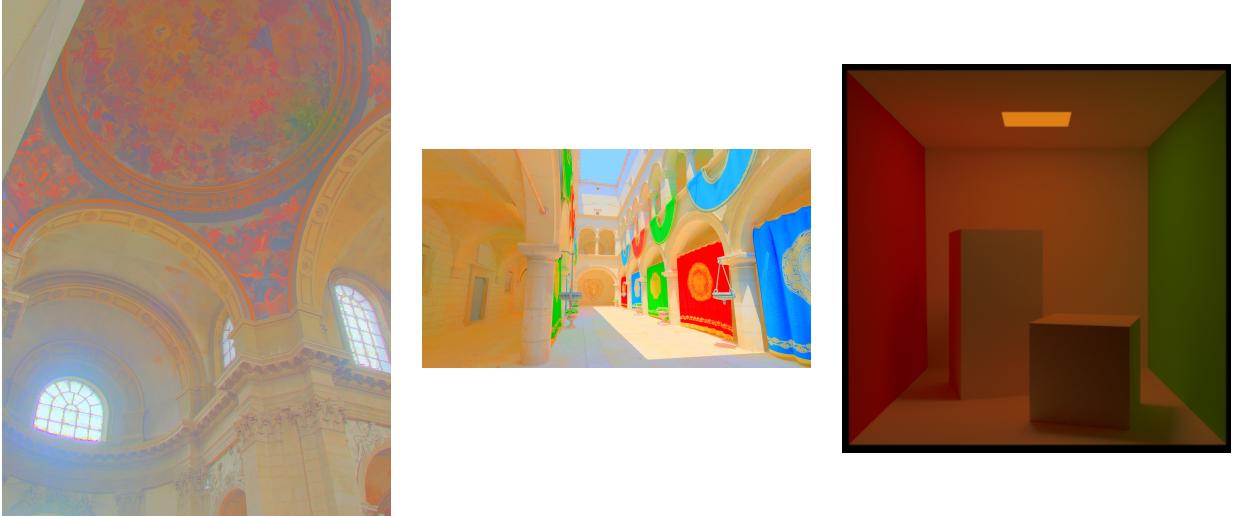


Figure 1: Schéma algoritmu převzato z článku Khan20 [1].

Funkce PQ vychází se snaží simulovat vnímaní lidského oka a transformuje vstupní jas v rozsahu  $[0, 10000] \text{ cd/m}^2$  na normalizovaný rozsah  $[0, 1]$ , přičemž nižší jasové hodnoty mají k dispozici větší relativní rozlišení.

Jak je vidět v tabulce 1, z takto transformovaného obrázku lze jasně vidět celou scénu, která ale působí vybledle a má relativně nízký kontrast.

Table 1: Obrázky po transformaci funkcí *Perceptual Quantizer*.



The proposed algorithm constructs histogram of the luminance of the PQ-transformed data, which is used to design the tone-mapping curve. In addition, a simple and effective method to handle the issues of excessive enhancement and compression of contrast is proposed. We perform detailed experimental evaluations using two well-known metrics that show a convincingly better performance of the proposed algorithm compared to the existing state of the art methods.

The authors however found that histogram-based strategy can overly compress some potentially important but smaller regions, and it can be overly generous to large regions which might not actually need many display levels. For example, a uniform blue sky covering half the image certainly does not require half of the total display levels, which a histogram-based strategy would allocate. The authors of [1] proposed a scheme that tries to solve this problem by keeping the pixel counts in histogram bins within defined lower and upper bounds.

Celý postup vypadá následovně:

- Transformace HDR obrazu pomocí PQ.
- Histogramová úprava kontrastu a binning.
- Transformace zpět do SDR.

## 4 Popis problému a řešení

Truncation and number of bins.

### 4.1 Podsekčí

Příklad obrázku, viz obr. 1.

## 5 Implementace

Popis implementace a vizualizace jednotlivých fází výpočtu.

## 6 Výsledky

See table 2 for comparison agains different methods. As it can be seen, on all testing images, the scene and all details are very clearly visible. I think the colors are sometimes bit too light and for more subjectively better looking images I'd stick with Drago03, as I like the darker and maybe a bit more saturated look of them. On the other hand, in terms of showing the details in darker regions I think this method is better. Also, I really like I don't have to tune parameters of the algorithm for each scene, because the defaults work across different scenes just fine.

Table 2: Comparison of different methods; first column is Khan20, second column Drago03 and third Ward94. All with default values of parameters that are set in Tone Mapping Studio.



Výpočetní složitost implementace je relativně nízká, program projde celkem 4x přes

všechny pixely obrázku a provede relativně jednoduché výpočty v každém kroku. Počet průchodů by šel snížit, nicméně takto je kód podle mého názoru přehlednější, kdy se každá funkce mapuje na jednu fázi algoritmu.

## 6.1 Uživatelská studie

Poznatky získané testováním.

# 7 Závěr

## References

- [1] KHAN, I. R., AZIZ, W., AND SHIM, S.-O. Tone-mapping using perceptual-quantizer and image histogram. *IEEE Access* 8 (2020), 31350–31358.
- [2] ČADÍK, M. Tone mapping studio. GitHub repository, 2025. Accessed: 2025-03-24.