

Rozhledy matematicko-fyzikální

Pavel Strachota

Transformace obrazu do frekvenční oblasti a jejich aplikace

Rozhledy matematicko-fyzikální, Vol. 98 (2023), No. 1, 35–55

Persistent URL: <http://dml.cz/dmlcz/151596>

Terms of use:

© Jednota českých matematiků a fyziků, 2023

Institute of Mathematics of the Czech Academy of Sciences provides access to digitized documents strictly for personal use. Each copy of any part of this document must contain these *Terms of use*.



This document has been digitized, optimized for electronic delivery and stamped with digital signature within the project *DML-CZ*:
The Czech Digital Mathematics Library <http://dml.cz>

Transformace obrazu do frekvenční oblasti a jejich aplikace

Pavel Strachota, FJFI ČVUT v Praze

Abstrakt. Tento článek představuje matematiku i aplikace diskrétní Fourierovy transformace ve formě přístupné středoškolským studentům. Motivací jsou algoritmy ztrátové komprese obrazu. Poskytujeme teoretický základ i vizuálně zajímavé praktické aplikace.

Úvod

Většina z nás dnes rutinně používá digitální fotoaparát v mobilu a pořizuje s ním fotografie a videa, která sdílí přes internet. Z internetu nebo ze svého vlastního datového úložiště sledujeme filmy a posloucháme hudbu. Zvykli jsme si říkat, že si posíláme ty „JPEGy“, posloucháme „MP-trojky“ a sledujeme „MP-čtyřky“ (pravda, to poslední se zase tak často neříká). Abychom to mohli dělat, musíme mít na počítači nainstalované ty správné „kodeky“ (na mobilu to zpravidla až tolik neřešíme, prostě to funguje).

Zkratky JPEG [1], MP3 [2], MP4 [3] a mnoho dalších podobných odkazují na formáty ukládání (kódování) audiovizuálních dat tak sofistikovaným způsobem, aby výsledné datové soubory zabíraly co nejméně místa, a přitom bylo možné z nich rekonstruovat (dekódovat) původní obrazový nebo zvukový záznam. Kombinací slov kódování a dekodování pak vznikl výraz „kodek“, označující implementaci těchto algoritmů, a to typicky ve formě programové knihovny využitelné více různými aplikacemi.

Oproti objemu původních dat jsou výsledné soubory někdy i o několik řádů menší, čehož lze na rozdíl od univerzálních kompresních algoritmů (např. známého ZIPu) dosáhnout použitím ztrátové komprese. To znamená, že rekonstruovaná data nejsou (z hlediska uchované informace, tj. hodnot jednotlivých bajtů) identická s daty původními, ale subjektivně na pohled, resp. na poslech člověk pozná jen nepatrný rozdíl. Dnešní algoritmy ztrátové komprese šikovně využívají vlastností lidského vnímání a z původních dat zachovávají jen takovou informaci, která je pro smysl člověka podstatná. Klíčovým nástrojem k dosažení tohoto cíle jsou matematické transformace, které umožní takzvaně „převést data z oblasti

prostorové do frekvenční“. Například u transformovaného obrázku již data neukazují, na kterém místě je která barva, ale z jak vysokých vln o různých frekvencích se obrázek skládá.

Cílem první části našeho článku je vysvětlit a matematicky dokázat princip těchto transformací a poté prakticky demonstrovat jejich funkci. U čtenáře nepředpokládáme vysokoškolské znalosti matematiky, a proto samotnému důkazu předchází shrnutí některých použitých pojmů a konceptů, které na první pohled s tématem přímo nesouvisí. V druhé části se vrátíme k algoritmům ztrátové komprese obrazu a ukážeme si, jakým způsobem se v nich využívají transformace do frekvenční oblasti. Nechybí praktické příklady demonstrující efekt ztrátové komprese na reálné obrázky.

Reprezentace rastrového obrazu

Obrazová nebo zvuková data lze matematicky vyjádřit jako funkci závislou na čase a prostorových souřadnicích. Například rovinný šedotónový (tj. „černobílý“) obdélníkový obraz je dán funkcí f , která pro každý bod (x, y) z tohoto obdélníku dodá číslo vyjadřující intenzitu (světlost) $f(x, y)$ v tomto bodě v rozsahu od 0 (černá) do 1 (bílá). V případě rastrového obrazu složeného z obrazových bodů (pixelů) je však situace jednodušší. Předpokládáme-li, že pixely mají čtvercový tvar a obrázek má velikost $N \times M$ pixelů, je funkce f po částech konstantní (s konstantní hodnotou na ploše každého pixelu) a může teoreticky¹⁾ nabývat nejvýše MN hodnot. Veškerou informaci o obrázku lze proto uložit do matice (tabulky čísel)

$$\mathbf{f} = \begin{pmatrix} f_{0,0} & f_{0,1} & \cdots & f_{0,N-1} \\ f_{1,0} & f_{1,1} & \cdots & f_{1,N-1} \\ \vdots & \vdots & & \vdots \\ f_{M-1,0} & f_{M-1,1} & \cdots & f_{M-1,N-1} \end{pmatrix}, \quad (1)$$

kde $f_{j,k}$ vyjadřuje intenzitu pixelu, který je v obraze j -tý shora a k -tý zleva, přičemž začínáme indexovat od nuly. Takto definovaná data lze transformovat do frekvenční oblasti pomocí dvourozměrné *diskrétní*

¹⁾Víme, že na počítači je i počet různých barev omezen. Např. v režimu True Color lze vybírat z celkem cca 16,7 milionu různých barev, ale pouze 256 z nich jsou odstíny šedi.

Fourierovy transformace (DFT)²⁾. Než si ukážeme jak, připomeneme si pár věcí z matematiky a trochu se procvičíme.

Součet N členů geometrické posloupnosti

Uvažujme posloupnost N čísel $a_0, a_1, a_2, \dots, a_{N-1}$. Jejich součet značíme

$$\sum_{n=0}^{N-1} a_n = a_0 + a_1 + a_2 \cdots + a_{N-1},$$

znak Σ (velké řecké písmeno sigma) se nazývá *suma* a písmeno n je *sčítací index*, který při sčítání dané N -tice čísel nabývá postupně hodnot $0, 1, 2, \dots, N - 1$.

Jednou z důležitých číselných posloupností je geometrická posloupnost, pro níž $a_n = q^n$ a q je dané reálné číslo, tzv. kvocient. Součet s_N prvních N členů geometrické posloupnosti najdeme snadno, pokud se nejprve zeptáme, kolik je $(1 - q) s_N$. Dostaneme totiž

$$\begin{aligned} (1 - q) s_N &= (1 - q) \sum_{n=0}^{N-1} q^n = \sum_{n=0}^{N-1} q^n - q \sum_{n=0}^{N-1} q^n = \sum_{n=0}^{N-1} q^n - \sum_{n=0}^{N-1} q^{n+1} \\ &= \sum_{n=0}^{N-1} q^n - \sum_{j=1}^N q^j = 1 - q^N. \end{aligned} \quad (2)$$

Úprava na druhém řádku v (2) spočívá v tzv. posunutí indexu, kdy definujeme $j = n + 1$, takže jestliže se index n pohyboval od 0 do $N - 1$, tak j půjde od 1 do N . V tu chvíli ale vidíme, že v obou sumách sčítáme pro $n = j = 1, 2, \dots, N - 1$ stejná čísla, která se navzájem odečtou. První suma má ovšem navíc ještě sčítanec $q^0 = 1$ (pro $n = 0$) a druhá má zase navíc q^N (pro $j = N$). Z této úvahy již plyne poslední rovnost v (2). Vyjádřením s_N z (2) pak pro $q \neq 1$ okamžitě dostáváme hledanou hodnotu

$$s_N = \frac{1 - q^N}{1 - q}, \quad (3)$$

čehož později využijeme.

²⁾ neboli Fourierovy transformace aplikované na *diskrétní data*, tj. hodnoty definované na konečné množině oddělených bodů. Mimo to existuje i (spojitá) Fourierova transformace, která transformuje funkce definované na množině všech reálných čísel.

Komplexní čísla

Rozumíme již tedy trochu sumám, ovšem stále jsme se nedostali k věci. Ještě jednou (ale slibuji, že už naposledy) se zdržíme, tentokrát u komplexních čísel. Popisovat užitečnost komplexních čísel je mimo rámec tohoto textu, ale při troše zjednodušení lze říci, že v oboru komplexních čísel mají řešení všechny kvadratické rovnice, včetně těch se záporným diskriminantem. Komplexní čísla jsou totiž ve tvaru

$$z = x + iy, \quad \text{kde } x, y \text{ jsou reálná čísla a } i^2 = -1.$$

Číslu i říkáme *imaginární jednotka*, x je reálná část a y je imaginární část komplexního čísla z . Číslo $\bar{z} = x - iy$ nazýváme číslo komplexně sdružené k číslu z .

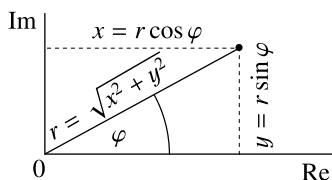
Každé komplexní z si lze také představit jako bod v rovině (tzv. *komplexní rovině*) o souřadnicích (x, y) , jak je vidět na obr. 1. Polohu tohoto bodu lze určit i v polárních souřadnicích, tj. jeho vzdáleností od počátku r a úhlem φ , který svírá jeho průvodič (spojnice s počátkem souřadné soustavy) s osou x (tzv. reálnou osou, protože čísla, pro která $y = 0$, jsou reálná). Zřejmě platí

$$x = r \cos \varphi,$$

$$y = r \sin \varphi,$$

$$r = \sqrt{x^2 + y^2}.$$

Číslu r říkáme *absolutní hodnota* komplexního čísla z (značíme $|z|$) a číslu φ *argument* komplexního čísla z .



Obr. 1: Komplexní číslo $z = x + iy$ jako bod v komplexní rovině a jeho polární souřadnice

Ukazuje se, že definice většiny elementárních funkcí reálné proměnné (např. $\sin x$, $\cos x$, e^x) lze vhodně rozšířit i do komplexního oboru. Pro tři

přávě zmíněné funkce a libovolné komplexní číslo z platí velmi důležitý vztah

$$e^{iz} = \cos z + i \sin z, \quad (4)$$

kterému se říká Eulerův vzorec. Díky němu můžeme psát

$$z = x + iy = r (\cos \varphi + i \sin \varphi) = re^{i\varphi},$$

a výrazu na pravé straně říkáme *goniometrický tvar* komplexního čísla z . Z něj je např. vidět, co se děje, když násobíme komplexní čísla $z_1 = r_1 e^{i\varphi_1}$ a $z_2 = r_2 e^{i\varphi_2}$. Dostáváme

$$z_1 z_2 = r_1 e^{i\varphi_1} r_2 e^{i\varphi_2} = r_1 r_2 e^{i(\varphi_1 + \varphi_2)},$$

takže součin má absolutní hodnotu rovnou součinu absolutních hodnot činitelů a argument rovný součtu argumentů jednotlivých činitelů³⁾.

Diskrétní Fourierova transformace

Konečně máme vše připraveno pro definici diskrétní Fourierovy transformace (DFT). Začneme pouze v jednorozměrném případě, kdy zdrojová data představují posloupnost N čísel f_0, f_1, \dots, f_{N-1} . Výsledkem DFT je jiná posloupnost N čísel F_0, F_1, \dots, F_{N-1} , tzv. *Fourierův obraz* nebo *spektrum* původní posloupnosti, jejíž n -tý prvek je definován vztahem

$$F_n = \sum_{k=0}^{N-1} f_k e^{-2\pi i k n / N} \quad \text{pro } n = 0, 1, \dots, N-1. \quad (5)$$

Čísla f_k mohou být reálná i komplexní, ale čísla F_n jsou obecně komplexní. Velmi důležité je, že ze znalosti Fourierova obrazu lze vždy získat původní posloupnost, a to pomocí inverzní DFT (iDFT), která má tvar

$$f_k = \frac{1}{N} \sum_{n=0}^{N-1} F_n e^{2\pi i k n / N} \quad \text{pro } k = 0, 1, \dots, N-1. \quad (6)$$

Toto tvrzení si snadno dokážeme. Zvolíme-li pevné k a dosadíme-li do pravé strany (6) tvar F_n ze vztahu (5), dostaneme

$$\frac{1}{N} \sum_{n=0}^{N-1} F_n e^{2\pi i k n / N} = \frac{1}{N} \sum_{n=0}^{N-1} \left(\sum_{j=0}^{N-1} f_j e^{-2\pi i j n / N} \right) e^{2\pi i k n / N}, \quad (7)$$

³⁾... z čehož pak plyne tzv. Moivreova věta $(\cos x + i \sin x)^n = \cos(nx) + i \sin(nx)$.

kde jsme museli zvolit jiné označení sčítacího indexu uvnitř velké závorky (j místo k), protože k už je pevně dané číslo. Obecně celkem N^2 sčítanců obsažených v tomto součtu lze sčítat v libovolném pořadí, což využijeme k úpravě (7) do tvaru

$$\sum_{j=0}^{N-1} f_j \frac{1}{N} \sum_{n=0}^{N-1} e^{-2\pi i j n / N} e^{2\pi i k n / N} = \sum_{j=0}^{N-1} f_j \frac{1}{N} \underbrace{\sum_{n=0}^{N-1} e^{2\pi i (k-j) n / N}}_{\text{OZL. } s_j}, \quad (8)$$

kde první (vnější) suma sčítá přes sčítací index j a vnitřní suma, kterou jsme označili jako s_j , sčítá přes n (v (7) to totiž bylo opačně). Navíc f_j bylo možné z vnitřní sumy vytknout, protože je to číslo nezávislé na n . Pro $j = k$ vnitřní suma obsahuje N jedniček ($e^0 = 1$), takže platí $s_k = N$. Pro $j \neq k$ si uvědomíme, že s_j je pouze součtem prvních N členů geometrické posloupnosti s kvocientem

$$q = e^{2\pi i (k-j) / N}.$$

Dle (3) tedy platí

$$s_j = \sum_{n=0}^{N-1} e^{2\pi i (k-j) n / N} = \frac{1 - q^N}{1 - q} = \frac{1 - e^{2\pi i (k-j)}}{1 - e^{2\pi i (k-j) / N}} = 0.$$

Poslední rovnost je vidět z Eulerova vzorce (4), neboť $(k - j)$ je celé číslo, a funkce \sin , \cos jsou periodické s periodou 2π , takže

$$e^{2\pi i (k-j)} = \cos(2\pi(k-j)) + i \sin(2\pi(k-j)) = 1 + i \cdot 0.$$

Tím jsme zjistili, že hodnota (8), což je hodnota pravé strany rovnosti (6), je skutečně rovna f_k .

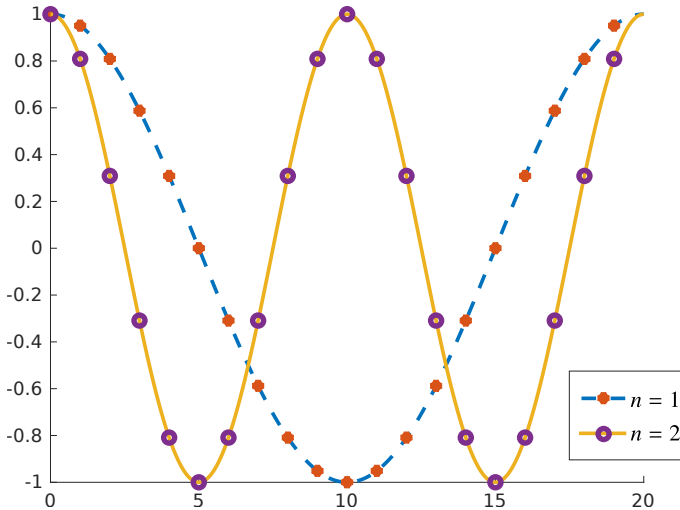
Abychom pochopili význam čísel F_n , vyjádříme si je v goniometrickém tvaru jako

$$F_n = r_n e^{i\varphi_n}$$

a dosadíme do definičního vztahu iDFT (6). Dostaneme

$$\begin{aligned} f_k &= \frac{1}{N} \sum_{n=0}^{N-1} r_n e^{i\varphi_n} e^{2\pi i k n / N} = \sum_{n=0}^{N-1} \frac{r_n}{N} e^{i(2\pi k n / N + \varphi_n)} \\ &= \sum_{n=0}^{N-1} \frac{r_n}{N} (\cos(2\pi k n / N + \varphi_n) + i \sin(2\pi k n / N + \varphi_n)). \end{aligned} \quad (9)$$

Z toho je vidět, že absolutní hodnoty čísel F_n (tj. čísla r_n) jsou až na faktor $1/N$ rovny amplitudám kosinových, resp. sinových vln, které jsou vzorkovány v bodech $(2\pi kn/N)$ pro $k = 0, 1, \dots, N - 1$. Argumenty čísel F_n (tj. čísla φ_n) pak definují jejich fázové posunutí. Sečtením všech takových vln vzniká původní posloupnost f_k . Přitom pro $n = 1$ projdeme pro $k = 0, 1, \dots, N - 1$ právě přes jednu periodu funkcí \cos , resp. \sin , pro $n = 2$ projdeme přes 2 periody atd. (viz obr. 2). Index n se proto nazývá *frekvence*.



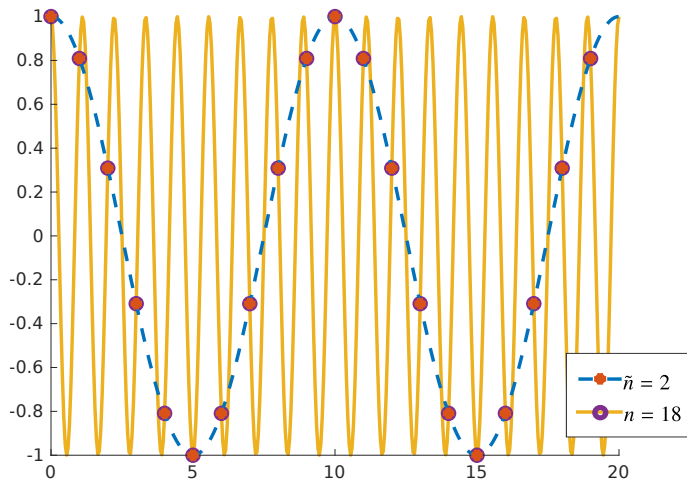
Obr. 2: Grafy funkce $\cos(2\pi xn/N)$ pro $N = 20$ a $n = 1, 2$ a její vzorky v N bodech, tj. hodnoty v bodech $x = k = 0, 1, \dots, N - 1$, které (když $\varphi_n = 0$) vystupují ve vztahu (9)

Je docela zajímavé pozorovat, co se děje pro vysoké frekvence, tj. když n je blízko počtu členů posloupnosti N . Definujme si $\tilde{n} = N - n$, kde $n > N/2$. Snadný výpočet ukáže, že platí

$$e^{2\pi i k n / N} = e^{2\pi i k (N - \tilde{n}) / N} = \underbrace{e^{2\pi i k N / N}}_{=1} e^{-2\pi i k \tilde{n} / N} = e^{-2\pi i k \tilde{n} / N} = \overline{e^{2\pi i k \tilde{n} / N}}. \quad (10)$$

V posledním kroku jsme využili Eulerův vzorec a definici komplexně sdruženého čísla. To znamená, že pro „velké“ $n > N/2$ má reálná část

výrazu $e^{2\pi ink/N}$ (tj. $\cos(2\pi ink/N)$) stejnou hodnotu jako pro „malé“ $\tilde{n} = N - n < N/2$, jak je vidět na obr. 3. Imaginární části pak mají navzájem opačná znaménka.



Obr. 3: Graf funkcí $\cos(2\pi xn/N)$ a $\cos(2\pi x\tilde{n}/N)$ pro $N = 20$, $n = 18$ a $\tilde{n} = N - n = 2$. I když jsou výsledné funkce evidentně odlišné, jejich vzorky v bodech $x = k = 0, 1, \dots, N - 1$ jsou stejné

Jestliže jsou navíc f_k reálná čísla (což při využití DFT pro transformace obrazu bude splněno), plyne z (10) a definičního vztahu (5) také vztah

$$F_n = \overline{F_{N-n}}. \quad (11)$$

Po aplikaci DFT na N reálných hodnot f_0, \dots, f_{N-1} tedy dostáváme (pro N sudé) jen $N/2$ nezávislých hodnot $F_0, \dots, F_{N/2-1}$ a zbytek lze dopočítat z (11). To není překvapující, protože F_n jsou komplexní a „vydají“ za dvě reálná čísla, takže v nich je dost informací na to, aby bylo možné původní N -tici rekonstruovat pomocí iDFT dle vztahu (6).

Spektrum digitálního obrazu a jeho úpravy

Nyní si ukážeme zajímavé aplikace DFT na obrazová data. Budeme potřebovat dvourozměrnou DFT, která vznikne dvěma aplikacemi jedno-rozměrné DFT na matici \mathbf{f} definovanou dle (1). Nejprve transformujeme

každý j -tý řádek obrazu dle

$$\tilde{F}_{j,n} = \sum_{k=0}^{N-1} f_{j,k} e^{-2\pi i kn/N}.$$

Poté aplikujeme DFT na výsledné hodnoty $\tilde{F}_{j,n}$ po sloupcích:

$$\begin{aligned} F_{m,n} &= \sum_{j=0}^{M-1} \tilde{F}_{j,n} e^{-2\pi i jm/M} = \sum_{j=0}^{M-1} \left(\sum_{k=0}^{N-1} f_{j,k} e^{-2\pi i kn/N} \right) e^{-2\pi i jm/M} \\ &= \sum_{j=0}^{M-1} \sum_{k=0}^{N-1} f_{j,k} e^{-2\pi i (kn/N + jm/M)}. \end{aligned} \quad (12)$$

Kdybychom postupovali nejprve po sloupcích a poté po řádcích, získali bychom totéž.

Zvolme si obrázek, na němž jsou vidět výrazné pravidelné pruhy, například ten na obr. 4a. Po aplikaci DFT si zobrazíme absolutní hodnoty spektra též jako obrázek, kde světlost jednotlivých bodů je úměrná hodnotám $|F_{m,n}|^4$). Kvůli symetrii (11), která nyní platí v obou rozměrech, je však šikovně přesunout pravou polovinu spektra doleva a dolní polovinu nahoru, takže dostaneme výsledek jako na obr. 4b, kde zastoupení nejnižších frekvencí ukazují body uprostřed. Hned si všimneme světlejších svislých pruhů. Ty dva nejbližší středu odpovídají frekvenci střídaní pruhů na stěně. Pruhy jsou ale ostře ohraničené, tj. jejich intenzita se nemění plynule jako hodnoty funkce sinus nebo kosinus. K získání ostřejšího barevného přechodu jsou totiž zapotřebí i vyšší frekvence, kterým odpovídají pruhy dále od středu. Abychom si tuto myšlenku ověřili, odstranili jsme všechny zmíněné pruhy ze spektra (příslušné hodnoty jsme nastavili na nulu), což je vidět na obr. 4d. Nakonec jsme provedli iDFT a výsledkem je obr. 4c⁵⁾. Pruhy skutečně z velké části zmizely, ale naše úpravy způsobily jako vedlejší efekt i částečné poškození zbytku obrázku.

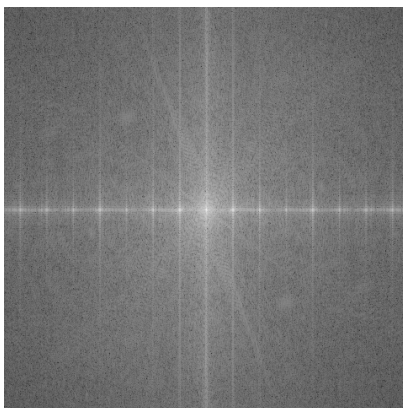
Pro demonstraci schopností DFT si ještě vyrobíme novou variantu optického klamu „Marilyn nebo Einstein“, která byla před časem velmi

⁴⁾Přesněji řečeno, světlost je na obr. 4b úměrná logaritmu $|F_{m,n}|$, abychom viděli jako dostatečně světlé i amplitudy vyšších frekvencí. Se zvyšováním frekvence totiž u každého „rozumného“ obrázku její amplituda prudce klesá, a to i o několik řádů.

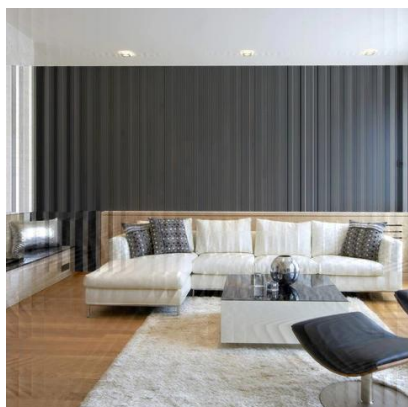
⁵⁾Pozornému čtenáři neujde, že obrázek je barevný, přičemž jsme se dosud bavili jen o šedotónových obrázcích. Barva ale není pro nás až taková věda. Stejně úpravy spektra jsme totiž provedli pro všechny tři barevné kanály R, G, B a výsledek pak opět složili dohromady.



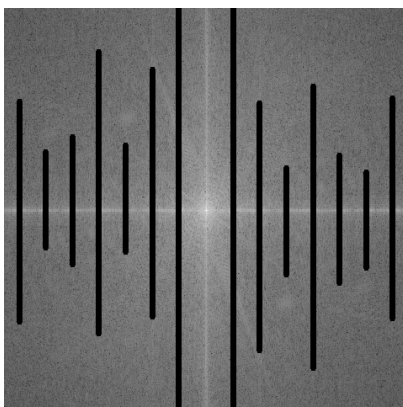
(a) Původní obrázek



(b) Spektrum původního obrázku



(c) Rekonstruovaný obrázek

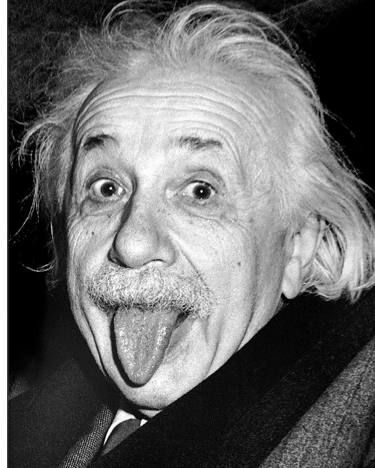


(d) Modifikované spektrum

Obr. 4: Odstranění některých frekvencí ve frekvenční oblasti a rekonstrukce obrázku z modifikovaného spektra

populární na internetu [4]. Jako zdrojová data nám poslouží fotografie obličejů Marilyn Monroe (obr. 5a) a Alberta Einsteina (5b), avšak záměrně jiné, než byly použity v původní verzi iluze. Po získání spektra fotografie Marilyn v něm vynulujeme amplitudy všech vyšších frekvencí mimo malý kroužek okolo středu (okolo nulové frekvence), neboli aplikujeme *nízkofrekvenční filtr*. Po provedení iDFT získáme obr. 5c. Je vidět, že z fotografie zmizely všechny detaily a je velmi „rozmazaná“. V pravém

horním rohu je pro představu zobrazena i aplikace nízkofrekvenčního filtru na spektrum, ovšem pro ilustraci je místo úplného vynulování použito jen ztmavení.

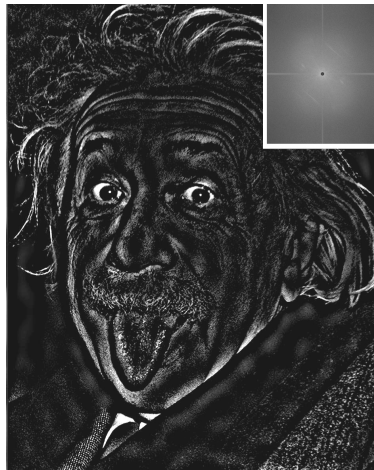


(a) Původní fotografie Marilyn Monroe

(b) Původní fotografie Alberta Einsteina



(c) Marilyn po aplikaci nízkofrekvenčního filtru



(d) Einstein po aplikaci vysokofrekvenčního filtru

Obr. 5: Příprava optického klamu „Marilyn nebo Einstein?“

Spektrum fotografie Einsteina zpracujeme zcela opačně. Aplikujeme *vysokofrekvenční filtr*, neboli vynulujeme jen malé kruhové okolí středu. Výsledek je na obr. 5d, kde naopak zůstaly kontury a tvary, ale zmizely barvy. Výslednou iluzi na obr. 6, pak získáme sečtením obou obrázků 5c a 5d. Na její princip přijdete snadno: pozorujeme-li obrázek z velké dálky (případně si sundáme brýle, případně si obrázek v PDF na počítači zmenšíme), detailní kontury nevidíme a zdá se nám, že na obrázku je Marilyn. Pokud se však podíváme dostatečně zblízka, zaujmou nás detailní kontury a na obrázku rozpoznáme Alberta Einsteina s vyplazeným jazykem.



Obr. 6: Výsledná podoba optického klamu „Marilyn nebo Einstein?“

Ztrátová komprese obrazu

Vraťme se nyní k algoritmům ztrátové komprese obrazu, kterými jsme naši diskusi začali. Vzhledem k tomu, o jak komplikovanou a rozsáhlou problematiku se dnes jedná, omezíme se na stručný a ne zcela přesný popis.

Komprese do více než 30 let starého a přitom stále populárního formátu JPEG začíná transformací barevných kanálů R, G, B, jejímž výsledkem je pro každý pixel informace o jeho světlosti (intenzitě) a o jeho odstínu. Vzhledem k tomu, že lidské oko mnohem snáze rozezná rozdíl v intenzitě než rozdíl v barevném odstínu, neukládají se hodnoty odstínu pro každý pixel samostatně, ale jen pro bloky o velikosti 2×2 pixely.

Poté nastupuje převod dat o intenzitách i o odstínech barev do frekvenční oblasti, a to samostatně a vždy po čtvercích velikosti 8×8 . Nepoužívá se přímo DFT, nýbrž tzv. diskretní kosinová transformace (DCT) [5], jejíž tvar lze odvodit aplikací DFT na data rozšířená pomocí vhodné symetrie (pro zájemce jsou technické detaily shrnuty v dodatku). Ve výsledku DCT pracuje jen s funkcí kosinus a výhodou je, že po transformaci reálných dat získáváme opět reálné koeficienty. Vlastnosti výsledku jsou však stejné, jako jsme si demonstrovali v předchozích experimentech s DFT: amplitudy kosinových vln velmi rychle klesají s frekvencí.

Ztrátovost komprese spočívá ve vhodném zaokrouhlení těchto amplitud, přičemž se spoléháme na to, že více či méně z nich se zaokrouhlí na nulu. Různou úroveň zaokrouhlování může uživatel řídit pomocí koeficientu kvality, který je vyjádřen v procentech. Z každého bloku 8×8 tak zbude jen několik málo nenulových hodnot, které se vhodně seřadí za sebe a podstoupí bezztrátovou kompresi pomocí dalších algoritmů, o nichž tento článek nepojednává. Při dekódování pomocí inverzní DCT se pak z malého počtu nenulových hodnot obnoví celý čtverec 8×8 , jehož hodnoty se pro „rozumné“ obrázky příliš neliší od originálu.

Výsledná úroveň komprese závisí na nastavení kvality i na složitosti obrázku, což si autor zkusil na svých nedávných fotografiích. Při rozlišení 5792×4344 pixelů (cca 25 megapixelů) a ukládání barevných hodnot R, G, B ve třech bytech pro každý pixel by každý takový obrázek zabíral v paměti cca 75 MB.



(a) kvalita 80%, velikost 4900 kB



(b) kvalita 80%, velikost 2840 kB

Obr. 7: Rozdílný kompresní poměr formátu JPEG v závislosti na obsahu snímku s rozlišením 5792×4344 pixelů

Fotografie venkovské krajiny se strništěm (obr. 7a), kde je k rozeznání

každé stéblo, má ve formátu JPEG s nastavením kvality na 80 % cca 5 MB, což je 1/15 původního objemu. Ovšem snímek městské zástavby (obr. 7b) má jen necelé 3 MB, což je dokonce 1/25 původního objemu dat.

V případě fotografií je obvykle nastavení kvality na 80 % dostatečné, neboť neumožňuje pouhým okem rozeznat jakýkoli rozdíl oproti nekomprimovanému obrazu. Abychom v tomto článku dokázali demonstrovat viditelné snížení kvality, uložili jsme výřez ze zmíněné fotografie krajiny s nastavením kvality na pouhých 5 %. Výsledek na obr. 8 ukazuje⁶⁾, že v mnoha případech (zejména na obloze) jsou bloky 8×8 vykresleny konstantní barvou. To znamená, že po zaokrouhlení koeficientů DCT zůstala v každém z těchto bloků jediná nenulová hodnota. Zároveň je značně redukován i počet barev.



(a) kvalita 98% (orig.), velikost 1100 kB



(b) kvalita 5%, velikost 64 kB

Obr. 8: Viditelná degradace fotografie venkovské krajiny při uložení JPEG s velmi nízkou kvalitou. Jedná se o výřez z fotografie na obr. 7a

Ve skutečnosti jsou však rozdíly oproti originálu přítomné pro jakékoliv nastavení kvality, což jsme se pokusili názorně ověřit pomocí volně dostupného grafického editoru GIMP⁷⁾. Uložili jsme fotografii na obr. 7a ve formátu JPEG s různým nastavením kvality a výsledek pak odečetli od originálu, což je rovněž JPEG s 98% kvalitou. Aby byly rozdíly lépe

⁶⁾Zobrazuje-li si čtenář tento článek v prohlížeči PDF souborů na obrazovce počítače, tabletu či mobilu, doporučujeme zvětšit si obr. 8 (a stejně tak i demonstrace na obr. 9, 11 a 12) a pozorně si prohlédnout detaily.

⁷⁾<https://www.gimp.org/>

viditelné, zvýraznili jsme je úpravou jasu a kontrastu (ve všech případech stejnou, aby bylo možné rozdíly porovnávat). Několik takto zpracovaných „rozdílových obrazů“ je vidět na obr. 9, který ukazuje, že se zvyšováním kvality dochází nejen ke zmenšování rozdílů, ale také k zjemňování struktury artefaktů ztrátové komprese.



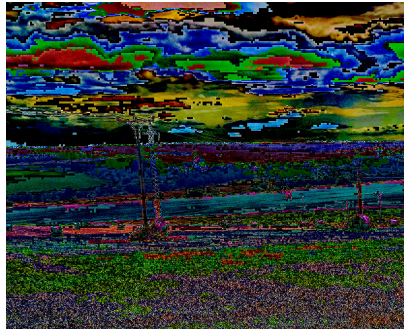
(a) kvalita 85%



(b) kvalita 50%



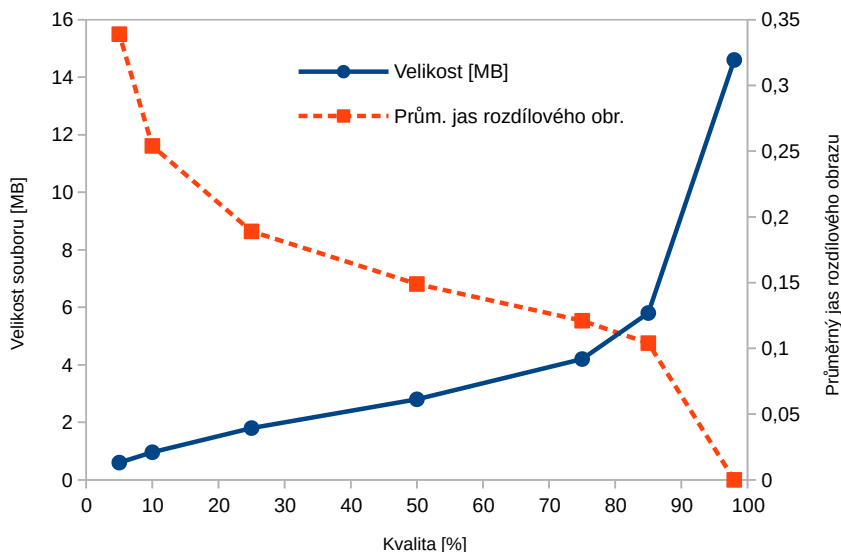
(c) kvalita 25%



(d) kvalita 5%

Obr. 9: Zvýrazněný rozdíl originálního obrazu a obrazu uloženého se ztrátovou kompresí JPEG pro různá nastavení kvality. Originální fotografie je na obr. 8a

Průměrná intenzita rozdílových obrazů společně s velikostí výsledných JPEG souborů je zanesena do grafu na obr. 10. Ten potvrzuje, že artefakty se zvyšující se kvalitou slábnou (ale jsou stále přítomny), zatímco velikost souboru roste. Pro nastavení kvality blížící se 100 % je nárůst velikosti souboru již velmi strmý.



Obr. 10: Závislost výskytu artefaktů ztrátové komprese (vyjádřeného průměrným jasnem rozdílového obrazu – viz obr. 9) a velikosti výsledného souboru na nastavení kvality JPEG. Referenčním obrazem je fotografie na obr. 7a, stažená z fotoaparátu ve formátu JPEG s kvalitou 98%

Jestliže se pokusíme do JPEGu uložit obraz obsahující ostré barevné přechody a velké jednobarevné plochy (např. kresbu na obr. 11), je možné artefakty ztrátové komprese pozorovat pouhým okem i při nastavení kvality na více než 75 %. Jak je vidět na obr. 12, rozdíly zcela nezmizí ani při kvalitě 98 %. Pro tento typ obrazu jsou vhodnější bezztrátové formáty, jako je PNG [6].

Ahoj, toto je test JPEG!

A toto je test s tenkými přechody...

(a) Originál (nekomprimovaný)

Ahoj, toto je test JPEG!

A toto je test s tenkými přechody...

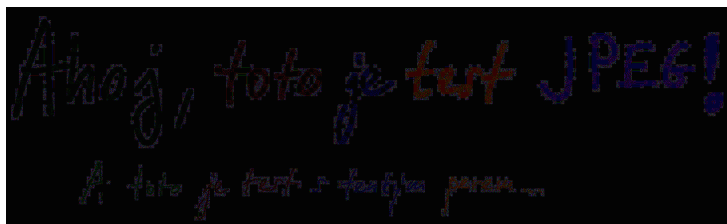
(b) kvalita 50%

Ahoj, toto je test JPEG!

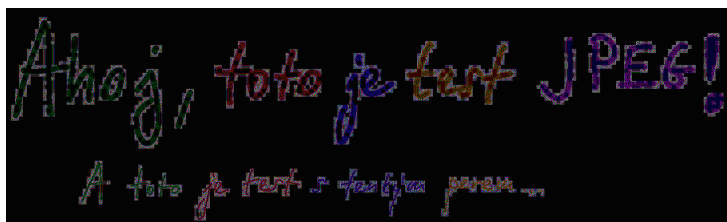
A toto je test s tenkými přechody...

(c) kvalita 10%

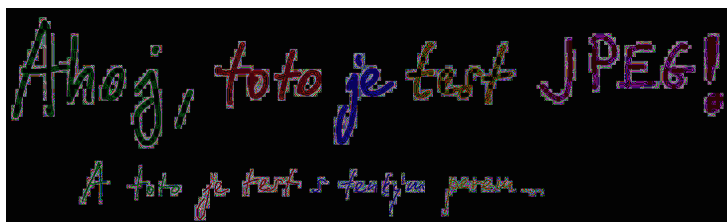
Obr. 11: Degradace obrazu s ostrými barevnými přechody při uložení do formátu JPEG



(a) kvalita 98%



(b) kvalita 85%



(c) kvalita 50%



(d) kvalita 10%

Obr. 12: Zvýrazněný rozdíl originálního obrazu a obrazu uloženého se ztrátovou kompresí JPEG pro různá nastavení kvality. Originální obraz s ostrými hranami je na obr. 11a

Snaha o ještě efektivnější kompresi zůstává aktuální i přes neustálé zvyšování rychlosti internetového připojení a kapacity úložišť, neboť zároveň roste i rozlišení snímků z digitálních fotoaparátů. Poslední dobou se objevily nové formáty ukládání obrazu, například formát HEIF, který využívá algoritmy komprese obrazu z video kodeku HEVC (H.265) [7], nebo WebP [8], který je zase spjatý s video kodekem VP9. I v nich však zůstává převod obrazu do frekvenční oblasti klíčovým krokem k efektivní ztrátové kompresi a stejně jako v JPEG se obraz zpracovává po čtvercových blocích, pro které se vžil název *makrobloky*.

Závěr

Matematické metody převodu dat do frekvenční oblasti založené na Fourierově transformaci a dalších příbuzných transformacích mají nezastupitelné místo nejen při zpracování obrazu a při ztrátové kompresi obrazu, videa a zvuku, ale též v telekomunikacích, při rekonstrukci třírozměrných dat z magnetické rezonance nebo v rentgenové a elektronové krystalografii. Svoje využití nalézají i v samotné matematice, například při hledání řešení parciálních diferenciálních rovnic. Cílem našeho článku bylo detailně představit jednu konkrétní metodu, tedy DFT, a to i zvidavým čtenářům se znalostmi matematiky na úrovni střední školy. Snažili jsme se proto vést výklad od základů a důsledně se vyhýbat konceptům matematické analýzy, jako jsou limity, nekonečné řady a integrály. Kdo se však těchto pojmů nebojí a zajímají ho širší souvislosti, může si prostudovat např. zdroje [9, 10], které jsou dostupné online. Kromě teoretického výkladu v článku nechybí ani vizuálně atraktivní praktické ukázky použití DFT a demonstrace následků ztrátové komprese obrazu.

Dodatek – diskrétní kosinová transformace

Pro čtenáře, kterým se zalíbila manipulace se sumami v kapitole o DFT, nabídneme jako bonus i odvození vzorce pro DCT, která je na DFT založená. V kompresi JPEG je použita dvourozměrná DCT typu II [1]. My odvodíme tvar příslušné jednorozměrné transformace, od níž lze ke dvojrozměrné variantě přejít stejným postupem jako v (12).

Z původní posloupnosti f_0, \dots, f_{N-1} připravíme celkem $4N$ hodnot x_0, \dots, x_{4N-1} tak, že za ni nejprve připojíme stejnou posloupnost v opačném pořadí, tj. $f_{N-1}, f_{N-2}, \dots, f_1, f_0$, a následně před každou původní hodnotu vložíme nulu, aby byly nenulové hodnoty pouze na lichých indexech. Pro výslednou posloupnost platí $x_{2k} = 0$, $x_{2k+1} = f_k$ pro

$k = 0, 1, \dots, N - 1, x_{2N} = 0$ a

$$x_\ell = x_{4N-\ell} \quad \text{pro } \ell = 2N + 1, 2N + 2, \dots, 4N - 1. \quad (13)$$

Na tyto hodnoty nyní aplikujeme DFT a postupnými úpravami dostaneme

$$\begin{aligned} X_n &= \sum_{k=0}^{4N-1} x_k e^{-2\pi i k n / (4N)} = \sum_{\substack{\text{jen přes} \\ \text{liché}}}^{2N-1} x_{2k+1} e^{-2\pi i (2k+1)n / (4N)} \\ &\stackrel{\text{úprava}}{=} \sum_{k=0}^{2N-1} x_{2k+1} e^{-\pi i (k+\frac{1}{2})n / N} \\ &\stackrel{\text{první}}{=} \sum_{k=0}^{N-1} x_{2k+1} e^{-\pi i (k+\frac{1}{2})n / N} + \sum_{k=N}^{2N-1} x_{2k+1} e^{-\pi i (k+\frac{1}{2})n / N} \\ &\stackrel{\text{druhá}}{=} \sum_{k=0}^{N-1} x_{2k+1} e^{-\pi i (k+\frac{1}{2})n / N} + \sum_{k=0}^{N-1} \underbrace{x_{2(2N-1-k)+1}}_{=x_{4N-(2k+1)}} e^{-\pi i ((2N-1-k)+\frac{1}{2})n / N} \\ &\stackrel{\text{druhá}}{=} \sum_{k=0}^{N-1} x_{2k+1} \left[e^{-\pi i (k+\frac{1}{2})n / N} + e^{-\pi i ((2N-1-k)+\frac{1}{2})n / N} \right] \\ &\stackrel{\text{úprava}}{=} \sum_{k=0}^{N-1} f_k \left[e^{-\pi i (k+\frac{1}{2})n / N} + \underbrace{e^{-2\pi i n}}_{=1} e^{+\pi i (k+\frac{1}{2})n / N} \right] \\ &\stackrel{\text{Eulerův}}{=} 2 \sum_{k=0}^{N-1} f_k \cos \left(\left(k + \frac{1}{2} \right) \frac{\pi n}{N} \right) = 2 \sum_{k=0}^{N-1} f_k \cos \left(\frac{(2k+1)\pi n}{2N} \right) \end{aligned} \quad (14)$$

pro $n = 0, 1, \dots, 4N - 1$. Protože X_n jsou zřejmě reálná čísla a zároveň platí (11), dostáváme

$$X_{4N-n} = X_n,$$

což umožňuje dopočítat $X_{2N+1}, \dots, X_{4N-1}$ z X_1, \dots, X_{2N-1} . Navíc využitím součtového vzorce $\cos(\alpha - \beta) = \cos \alpha \cos \beta + \sin \alpha \sin \beta$ lze snadno ukázat, že

$$X_{2N-n} = -X_n, \quad (15)$$

což umožňuje získat X_{N+1}, \dots, X_{2N} z X_0, \dots, X_{N-1} . Nakonec z (15) plyne ještě $X_N = 0$. Celkem tedy z (14) získáváme nezávislé hodnoty X_n pouze pro $n = 0, 1, \dots, N-1$. Za výsledek DCT posloupnosti f_0, \dots, f_{N-1} se proto považuje posloupnost N čísel X_0, \dots, X_{N-1} .

Literatura

- [1] Pennebaker, W. B., Mitchell, J. L.: *JPEG Still Image Data Compression Standard*. Springer, New York, 1993.
- [2] Hacker, S.: *MP3: The Definitive Guide*. O'Reilly Media, 2000.
- [3] Wootton, C.: *A Practical Guide to Video and Audio Compression: From Sprockets and Rasters to Macro Blocks*. Routledge, 2005.
- [4] Zolfagharifard, E.: *Who do YOU see: Albert or Marilyn? Optical illusion could reveal if you need glasses*. *Daily Mail Online*, č. 3. dubna 2015, Dostupné na: <https://www.dailymail.co.uk/sciencetech/article-3024005/Do-Albert-Einstein-Marilyn-Monroe-Hybrid-optical-illusion-reveal-need-wear-glasses.html> [cit. 16.8.2022].
- [5] Tišnovský, P.: *Programujeme JPEG: diskrétní kosinová transformace (DCT)*. *ROOT.cz*, č. 4. ledna 2007, Dostupné na: <https://www.root.cz/clanky/programujeme-jpeg-diskretni-kosinova-transformace-dct/> [cit. 16.8.2022].
- [6] Roelofs, G.: *PNG—The Definitive Guide*. O'Reilly, 1999.
- [7] Elemental Technologies: *HEVC Demystified—A Primer on the H.265 Video Codec*. The Broadcast Bridge, 2014.
- [8] Google: *An Image Format for the Web*. Dostupné na: <https://developers.google.com/speed/webp> [cit. 16.8.2022].
- [9] Horák, D.: *Diskrétní transformace*. ZČU, Plzeň, 2012, učební text. Dostupné na https://mi21.vsb.cz/sites/mi21.vsb.cz/files/unit/diskretni_transformace.pdf [cit. 16.8.2022].
- [10] Hlaváč, V.: *Fourierova transformace v 1D a 2D*. ČIRK ČVUT, Praha, dostupné na: <https://people.ciirc.cvut.cz/~hlavac/TeachPresCz/11DigZprUbr/12FourierTxCz.pdf> [cit. 16.8.2022].