

小波分析作业

于冰冰 21901037 数硕 1903

2020 年 1 月 29 日

目录

1	用 Haar 小波对图像进行分解和重构	2
1.1	简介	2
1.2	实验结果	2
2	小波神经网络	3
2.1	简介	3
2.2	实验结果	3
2.2.1	NN 实验结果	3
2.2.2	Morlet MNN 实验结果	4
2.2.3	POLYWOG3 WNN 实验结果	5
3	小波调制	7
4	$L^2_{\omega}(-1, 1)$ 空间的多项式小波分解	9
5	Discrete Lattice Wavelet Transform	10
6	提升小波	12
7	参考文献	14

1 用 Haar 小波对图像进行分解和重构

1.1 简介

小波变换理论具有许多其它理论所不具备的优良特性，如正交性、方向选择性、可变的时（空）频域分辨率等，这些良好的分析特性使小波变换成为信号处理的一种强有力的工具和手段。本文使用 Haar 小波对图像进行分解和重构。

小波分解的基本方法是讲图片分别通过低通滤波器和高通滤波器，重构的基本方法是使通过低通滤波器和高通滤波器之后的图片分别通过重构后的低通滤波器和高通滤波器。同时在变换过程中可以进行阈值去噪。

小波阈值去噪的基本思想:Donoho 提出的小波阈值去噪的基本思想是将信号通过小波变换（采用 Mallat 算法）后，信号产生的小波系数含有信号的重要信息，将信号经小波分解后小波系数较大，噪声的小波系数较小，并且噪声的小波系数要小于信号的小波系数，通过选取一个合适的阈值，大于阈值的小波系数被认为是信号产生的，应予以保留，小于阈值的则认为是噪声产生的，置为零从而达到去噪的目的。其基本步骤为：

1. 分解：选定一种层数为 N 的小波对信号进行小波分解；
2. 阈值处理过程：分解后通过选取一合适的阈值，用阈值函数对各层系数进行量化；
3. 重构：用处理后的系数重构信号

1.2 实验结果

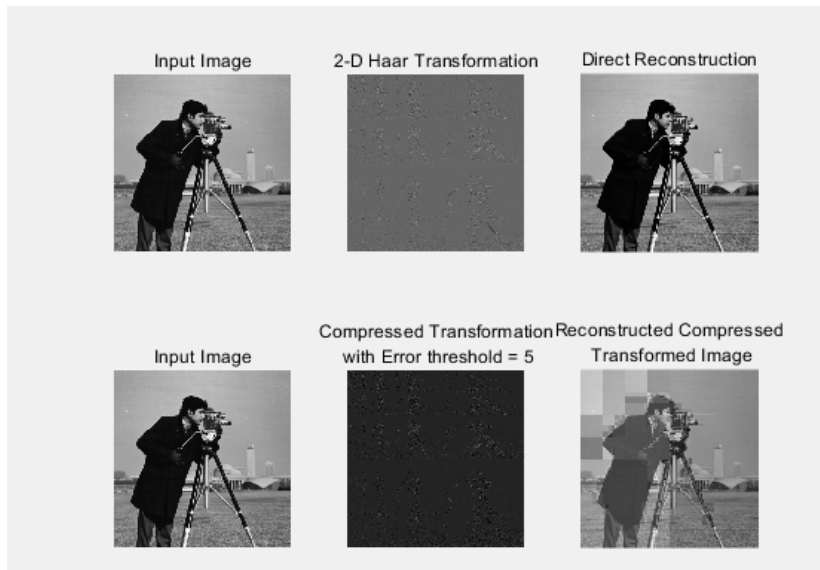


图 1: haar 小波分解和重构结果

2 小波神经网络

2.1 简介

小波神经网络沿用了神经网络的结构,并使用小波函数作为激活函数。在程序中,我们先使用 sigmoid 作为激活函数构造神经网络,然后使用小波函数 morlet 函数和小波函数 POLYWOG3 函数作为激活函数,对比其实验效果。实验为对 MNIST 数据集中的数字进行识别

神经网络 (Neural Network, NN) 程序实现包含 2 个隐藏层的神经网络,激活函数为 sigmoid 函数。

小波神经网络 (Wavelet Neural Network, WNN) 程序实现包含 1 个小波隐藏层的小波神经网络,激活函数分别选为小波函数 morlet 函数和小波函数 POLYWOG3 函数。

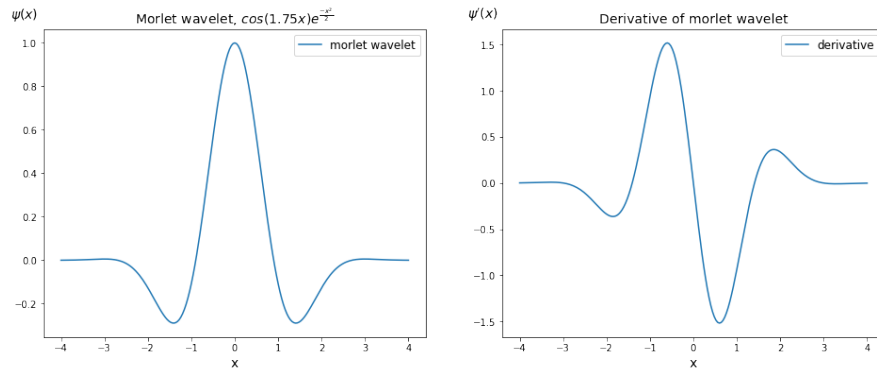


图 2: morlet 函数的图像 (左) 及其导数图像 (右)

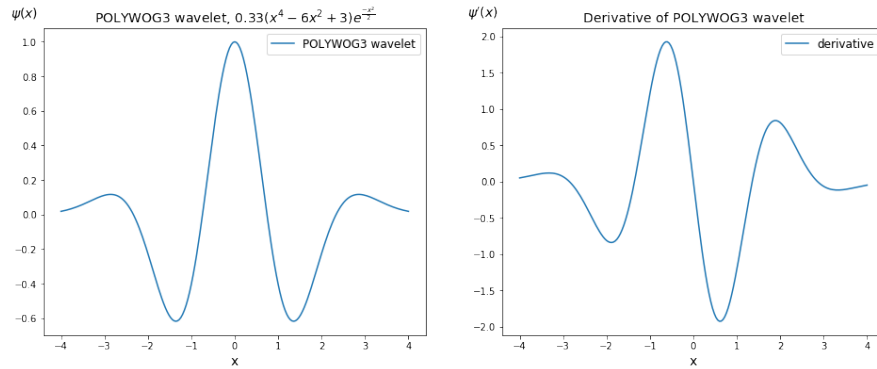


图 3: polywog3 函数的图像 (左) 及其导数图像 (右)

2.2 实验结果

当激活函数选为 morlet 函数时,单个隐层的小波神经网络的能力与双个隐层的神经网络相当。当激活函数选为 polywog3 函数时,网络收敛速度明显快于普通的神经网络,精度在经过 40 次迭代之后达到了含双隐层的普通神经网络需要 200 次迭代才能达到的结果。

2.2.1 NN 实验结果

训练结果:

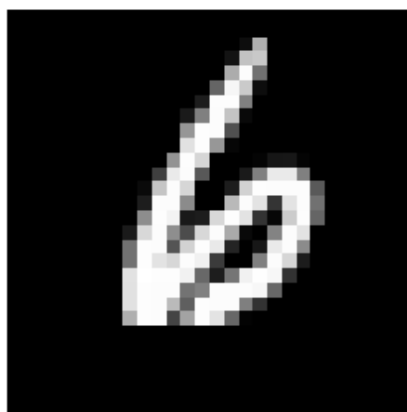
Epoch 1, mse_loss: 0.3225, accury on the training set :52.12%

Epoch 20, mse_loss: 0.0664, accury on the training set :88.18%
 Epoch 40, mse_loss: 0.0484, accury on the training set :94.68%
 Epoch 60, mse_loss: 0.0127, accury on the training set :97.90%
 Epoch 80, mse_loss: 0.0101, accury on the training set :98.26%
 Epoch 100, mse_loss: 0.0090, accury on the training set :98.38%
 Epoch 120, mse_loss: 0.0086, accury on the training set :98.42%
 Epoch 140, mse_loss: 0.0081, accury on the training set :98.52%
 Epoch 160, mse_loss: 0.0077, accury on the training set :98.58%
 Epoch 180, mse_loss: 0.0073, accury on the training set :98.64%
 Epoch 200, mse_loss: 0.0073, accury on the training set :98.64%
 Training time is :1080.96 s.

测试结果:

测试准确率 90.8

```
In [0]: plt.imshow(training_data[0][10].reshape(28,-1), 'gray')
plt.xticks([], plt.yticks([]))
plt.show()
```



```
In [0]: training_data[1][10]
```

```
Out[0]: 6
```

```
In [0]: net.predict(training_data[0][10])
```

```
Out[0]: array([6], dtype=uint8)
```

图 4: NN 实验结果

2.2.2 Morlet MNN 实验结果

训练结果:

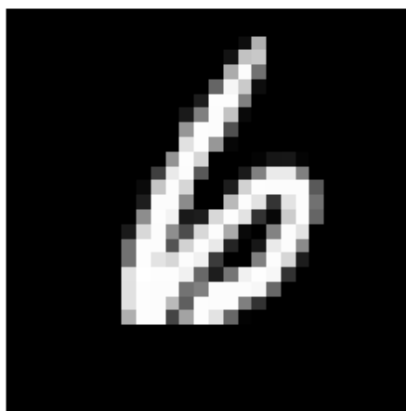
Epoch 1, mse_loss: 0.3225, accury on the training set :52.12%
 Epoch 20, mse_loss: 0.0664, accury on the training set :88.18%
 Epoch 40, mse_loss: 0.0484, accury on the training set :94.68%

Epoch 60, mse_loss: 0.0127, accury on the training set :97.90%
 Epoch 80, mse_loss: 0.0101, accury on the training set :98.26%
 Epoch 100, mse_loss: 0.0090, accury on the training set :98.38%
 Epoch 120, mse_loss: 0.0086, accury on the training set :98.42%
 Epoch 140, mse_loss: 0.0081, accury on the training set :98.52%
 Epoch 160, mse_loss: 0.0077, accury on the training set :98.58%
 Epoch 180, mse_loss: 0.0073, accury on the training set :98.64%
 Epoch 200, mse_loss: 0.0073, accury on the training set :98.64%
 Training time is :1080.96 s.

测试结果:

测试准确率 90.2

```
In [0]: plt.imshow(training_data[0][10].reshape(28,-1), 'gray')
        plt.xticks([], plt.yticks([])
        plt.show()
```



```
In [0]: training_data[1][10]
```

```
Out[0]: 6
```

```
In [0]: net.predict(training_data[0][10])
```

```
Out[0]: array([6], dtype=uint8)
```

图 5: Morlet MNN 实验结果

2.2.3 POLYWOG3 WNN 实验结果

训练结果:

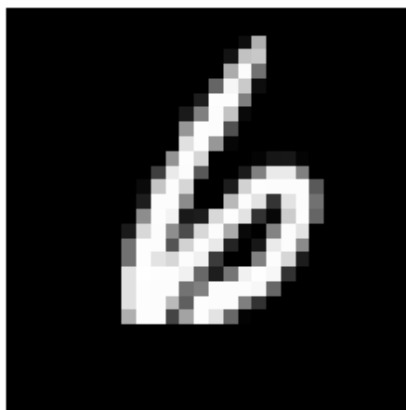
Epoch 1, mse_loss: 0.3483, accury on the training set :48.18%
 Epoch 20, mse_loss: 0.0474, accury on the training set :93.06
 Epoch 40, mse_loss: 0.0246, accury on the training set :95.72%
 Epoch 60, mse_loss: 0.0198, accury on the training set :96.48%

Epoch 80, mse_loss: 0.0170, accury on the training set :96.86% Epoch 100, mse_loss: 0.0160, accury on the training set :96.96% Epoch 120, mse_loss: 0.0153, accury on the training set :97.12% Epoch 140, mse_loss: 0.0143, accury on the training set :97.28% Epoch 160, mse_loss: 0.0139, accury on the training set :97.38% Epoch 180, mse_loss: 0.0135, accury on the training set :97.44% Epoch 200, mse_loss: 0.0133, accury on the training set :97.54% Training time is :1207.57 s.

测试结果:

测试准确率: 93.6

```
In [0]: plt.imshow(training_data[0][10].reshape(28,-1), 'gray')
plt.xticks([], plt.yticks([]))
plt.show()
```



```
In [0]: training_data[1][10]
```

```
Out[0]: 6
```

```
In [0]: net.predict(training_data[0][10])
```

```
Out[0]: array([6], dtype=uint8)
```

图 6: POLYWOG3 MNN 实验结果

3 小波调制

我们知道，小波中的 Haar 尺度函数表达式为：

$$\phi_H(t) = \begin{cases} 1 & t \in [0, 1] \\ 0 & t \notin [0, 1] \end{cases} \quad (1)$$

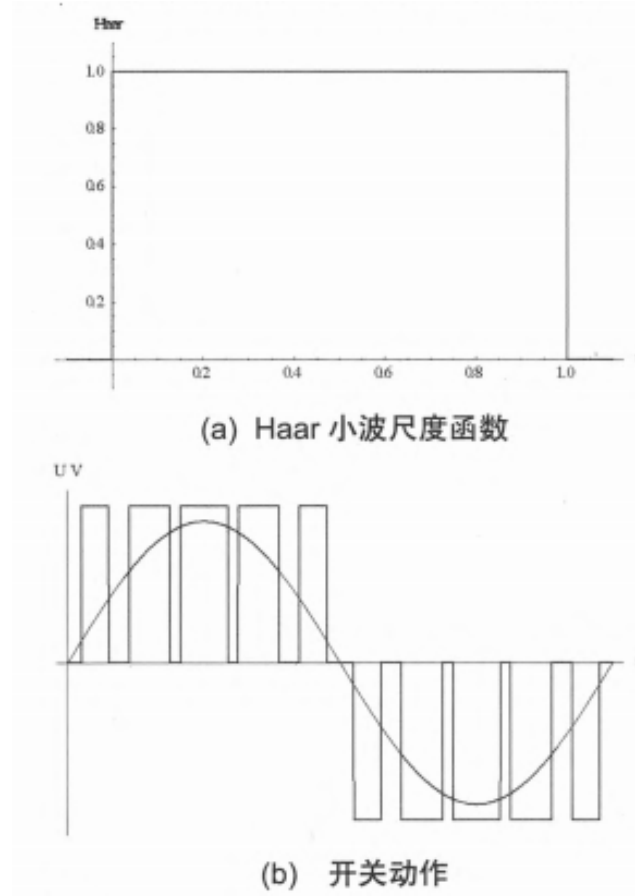


图 7: 小波调制示意图

比较图 7 的 a 和 b 可知，逆变器的开关过程可以看成是 Haar 小波尺度函数经过伸缩和平移变换后再组合而成的结果。基于这个事实，Saleh 和 Rahman 提出了一种用于逆变器的基于非均匀采样的数学模型，该模型将逆变器的开关信号的产生看作是一种非均匀采样和重构的过程。利用 Haar 小波尺度函数构造的线性合成尺度函数为：

$$\phi_j(t) = \phi_H(2^{j+1}t) + \phi_H(2^{j+1}(t-1+2^{-(j+1)})) \quad (2)$$

其中 $j = 0, 1, 2, \dots$, $\phi_H(t)$ 为 Haar 小波尺度函数。该线性合成尺度函数支持非均匀周期结构的采样。

与线性合成尺度函数对应的合成尺度函数为：

$$\tilde{\varphi}_j(t) = (\phi_H)_j(t) - \psi_j(t) \quad (3)$$

小波调制法用 $\tilde{\varphi}_j(t)$ 模拟不同宽度的矩形脉冲，矩形脉冲的宽度随着 j 的增大而增大。

由小波分析的知识我们可以知道，一个连续时间信号 $X_c(t)$ 可以利用线性合成尺度函数表示为：

$$x_\epsilon(t) = \sum_j \sum_{j \in Z} \langle x_\epsilon(t), \varphi_j(t-d) \rangle \tilde{\varphi}_j(t-d) \quad (4)$$

其中 j, k 为整数。这个式子中，内积 $\langle x_c(t), \phi_j(t-d) \rangle$ 表示对连续时间信号 $x_c(t)$ 进行非均匀采样， $\sum_j \sum_{d \in Z} \langle x_c(t), \varphi_j(t-d) \rangle \tilde{\varphi}_j(t-d)$ 表示对采样得到的值进行的重构。

用于单项逆变器小波调制的实现步骤为：

1. 设定尺度参数 j 的初始值 $j_0 = 1$ ，采样组序 d 的初始值；
2. 设定一个采样点为 $t_{d1} = d + 2^{-j-1}$ ，另一个采样点为 $t_{d2} = d + 1 - 2^{-j-1}$ ；
3. 在上述两个采样点内形成一个开关脉冲；
4. 计算 $S'_M(t_{d2})$ 的值， $S'_M(t_{d2}) = \omega_m \cdot \cos(\omega_m t_{d2})$ ，如果 $S'_M(t_{d2}) > 0$ ，则下一组采样组 j 加 1；如果 $S'_M(t_{d2}) < 0$ ，则下一组采样组 j 减一；
5. 如果 $t \geq T$ ，则 $j = j_0, d = d_0$ ，否则 $d = d + 1$ ；
6. 返回第二步，重新计算

4 $L^2_\omega(-1, 1)$ 空间的多项式小波分解

设 $\omega(x) = (1+x)^{1/2}(1-x)^{-1/2}$, 并设 $L^2_\omega(-1, 1)$ 关于权函数 $\omega(x)$ 平方可积, 函数空间

$$L^2_\omega(-1, 1) = \left\{ f(x) : \int_{-1}^1 |f(x)|^2 \omega(x) dx < \infty \right\} \quad (5)$$

如通常, 定义 $f, g \in L^2_\omega(-1, 1)$ 的内积为 $(f, g) := \int_{-1}^1 f(x)g(x)\omega(x)dx$.

再记 Π_n 为次数不超过 n 的多项式之集. Jacobi 多项式 $P_n^{(\alpha, \beta)}(x)$ 通过下面微分方程定义:

$$(1-x)^{\alpha} (1+x)^{\beta} P_n^{(\alpha, \beta)}(x) = \frac{(-1)^n}{2^n n!} \frac{d^n}{dx^n} ((1-x)^{n+\alpha} (1+x)^{n+\beta}) \quad (6)$$

关于权函数 $\omega(x)$ 正交的多项式时 jacobi 多项式 $P_n^{(-\frac{1}{2}, \frac{1}{2})}(x)$

$$P_n^{(-\frac{1}{2}, \frac{1}{2})}(x) = \frac{1 \cdot 3 \cdots (2n-1)}{2 \cdot 4 \cdots 2n} \frac{\cos\left(\frac{2n+1}{2} \arccos x\right)}{\cos\left(\frac{1}{2} \arccos x\right)} =: \frac{1 \cdot 3 \cdots (2n-1)}{2 \cdot 4 \cdots 2n} J_n(x) \quad (7)$$

定义尺度函数

$$\phi_{j,l}(x) := \frac{\omega_j(x)}{\omega'_j(\cos \theta_{j,2l})(x - \cos \theta_{j,2l})} \quad l = 0, 1, \dots, 2^j \quad (8)$$

其中

$$\theta_{j,l} = \frac{l\pi}{2^{j+1} + 1} \quad (9)$$

定义尺度空间 V_j 为

$$V_j := \text{span} \{ \phi_{j,l} : l = 0, 1, \dots, 2^j \} \quad (10)$$

容易看出

$$V_j = \Pi_2^j, \dim V_j = \dim \Pi_{2^j} = 2^j + 1 \quad (11)$$

而且

$$V_0 \subset V_1 \subset V_2 \subset \cdots \subset V_j \subset \cdots \quad (12)$$

$$\overline{\bigcup V_j} = L^2_\omega(-1, 1) \quad (13)$$

记

$$\psi_{j,l}(x) = \frac{2C_{i+2}(1 + \cos \theta_{i,2+1})J_{2'}(x)}{(C_{2j+2} + 3 \cdot 2^j)(x - \cos \theta_{j,2l+1})} \left(\frac{C_{i+1}}{C_{j+2}}(1-x)U_2'(x) - (-1)^l \frac{C_j}{C_{j+1}} \left(\sin \frac{1}{2} \theta_{j,2l+1} \right) \right) \quad (14)$$

其中 $C_j = 2^j + 1$

定义

$$W_j := \text{span} \{ \psi_{j,l}, l = 0, 1, \dots, 2^j - 1 \} \quad (15)$$

则有 $\dim W_j = 2^j, W_j \subset \Pi_{2^{j+1}}$

并且小波满足分解:

$$V_{j+1} = V_j \oplus W_j \quad (16)$$

证明过程在此省略。

5 Discrete Lattice Wavelet Transform

在信号处理领域，离散小波变换 (DWT) 已经获得了广泛应用，在许多理论论文中，正交镜像滤波器 (QMF) 和共轭正交滤波器 (CQF) 是低通滤波器和高通滤波器的替代方案。然而，QMF 滤波器中只有 Haar 小波才具有完美重构属性 (PR)；CQF 滤波器具有 PR 属性，但缩放滤波器和小波滤波器没有对称系数。

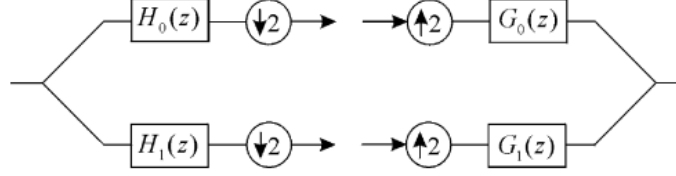


图 8: Analysis and synthesis parts of the two-channel DWT.

双通道的离散小波变换结构如图 8 所示，在这种结构中，低通滤波器的频率特性 $H_0(z)$ 和高通小波滤波器 $H_1(z)$ 和二项式项 $(1 + z^{-1})^K$ 相关，这个多项式有 K 个根 $z = \mp 1$ ，这种低通滤波器和高通滤波器的性质相对较差。

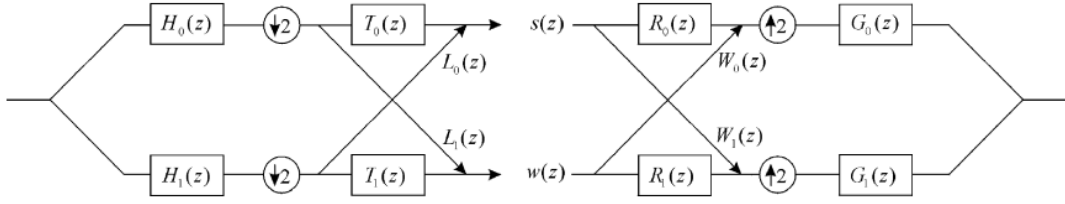


图 9: General structure for the two-channel DLWT

双通道的 DLWT 结构如图 9 所示，它包含尺度缩放 $H_0(z)$ 、小波滤波器 $H_1(z)$ 和 Lattice 网络。并行通道中含有转换滤波器 $T_0(z)$ 和 $T_1(z)$ ，他们通过两个交叉的 lattice $L_0(z)$ 和 $L_1(z)$ 交换信息。在合成部分，lattice 由两个透射滤波器 $R_0(z)$ 和 $R_1(z)$ ，还有两个交叉滤波器 $W_0(z)$ 和 $W_1(z)$ ，以及最后两个重建滤波器 $G_0(z)$ 和 $G_1(z)$ 组成

我们需要 DLWT 满足 PR 属性，即输出应和输入信号相同。假定缩放、小波和重构滤波器满足 PR 属性：

$$\begin{aligned} H_0(z)G_0(z) + H_1(z)G_1(z) &= 2z^{-k} \\ H_0(-z)G_0(z) + H_1(-z)G_1(z) &= 0 \end{aligned} \quad (17)$$

Lattice 网络 (图 10) 的 PR 条件为

$$\begin{bmatrix} y_0(z) \\ y_1(z) \end{bmatrix} = \begin{bmatrix} z^{-k} & 0 \\ 0 & z^{-k} \end{bmatrix} \begin{bmatrix} x_0(z) \\ x_1(z) \end{bmatrix} \quad (18)$$

其中 $x_0(z)$ 和 $x_1(z)$ 是第 z 次转换的输入信号， $y_0(z)$ 和 $y_1(z)$ 是第 z 次转换的输出信号，通常情况下有

$$\begin{bmatrix} y_0(z) \\ y_1(z) \end{bmatrix} = \begin{bmatrix} T_0R_0 + L_1W_0 & L_0R_0 + T_1W_0 \\ T_0W_1 + L_1R_1 & T_1R_1 + L_0W_1 \end{bmatrix} \begin{bmatrix} x_0(z) \\ x_1(z) \end{bmatrix} \quad (19)$$

如果 L 我们声明 attice 网络矩阵的反对角元素为 0，即

$$\begin{aligned} W_0 &= -L_0 \\ W_1 &= -L_1 \\ R_0 &= T_1 \\ R_1 &= T_0 \end{aligned} \quad (20)$$

那么我们得到 PR 条件：

$$T_0(z)T_1(z) - L_0(z)L_1(z) = z^{-k} \quad (21)$$

对于 DLWT 结构来说，有几种方法可以设计 Lattice 滤波器为【使其遵循 PR 条件。最一般的情况为：对于 k 为奇数，我们定义一个多项式

$$P(z) = T_0(z)T_1(z) = \left[B(z^2) + \frac{1}{2}z^{-k} \right] \quad (22)$$

其中 $B(z^2)$ 为一个 z^2 的多项式，此时如果有

$$\begin{aligned} L_0(z) &= T_1(-z) \\ L_1(z) &= T_0(-z) \end{aligned} \quad (23)$$

则 PR 条件满足。并且最终的 PR 条件为

$$P(z) - P(-z) = z^{-k} \quad (24)$$

当 $P(z)$ 满足形式：

$$P(z) = z^{-k} \left[\frac{1}{2} + a_1(z + z^{-1}) + a_3(z^3 + z^{-3}) + \cdots + a_n(z^n + z^{-n}) \right] \quad (25)$$

该形式具有频率响应：

$$P(e^{j\omega}) = e^{-jk\omega} \left[\frac{1}{2} + 2 \sum_{\text{odd } i=1}^n a_i \cos(i\omega) \right] \quad (26)$$

在实际过程中， $P(z)$ 可以通过 Parks-McClellan 算法进行设计。频率响应 $P(e^{j\omega})$ 是基于 $\pi/2$ 完全对称的然后可以基于频谱分解的方法将 $P(z)$ 分解为 $T_0(z)$ 和 $T_1(z)$ 。随着 n 的增加，有多种形式的 $P(z)$ 满足条件。

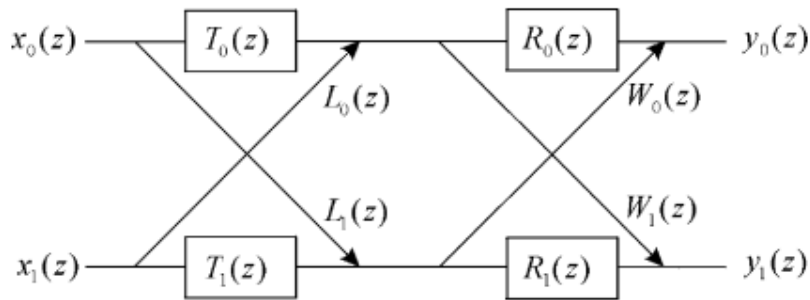


图 10: General structure for the two-channel DLWT

6 提升小波

二维离散小波变换最有效的实现方法之一是采用 Mallat 算法, 通过在图像的水平和垂直方向交替采用低通和高通滤波实现, 如图 11所示. 这种传统的基于卷积的离散小波变换计算量大, 计算复杂度高, 对存储空间的要求高, 不利于硬件实现. 提升小波的出现有效地解决了这一问题. 提升算法相对于 Mallat 算法而言, 是一种更为快速有效的小波变换实现方法, 被誉为第 2 代小波变换. 它不依赖于傅立叶变换, 继承了第 1 代小波的多分辨率特征, 小波变换后的系数是整数, 计算速度快, 计算时无需额外的存储开销. Daubechies 已经证明, 任何离散小波变换或具有有限长滤波器的两阶滤波变换都可以被分解成为一系列简单的提升步骤, 所有能够用 Mallat 算法实现的小波, 都可以用提升算法来实现.

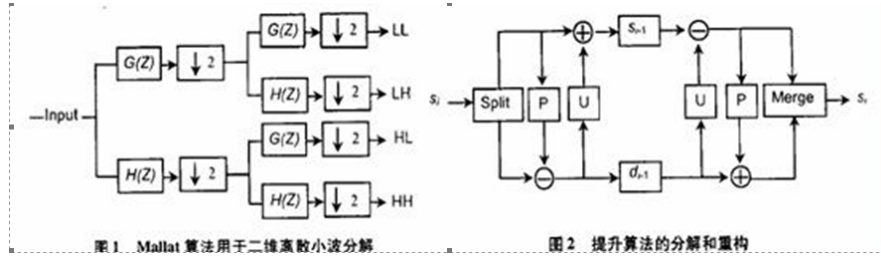


图 11: 提升算法

对信号进行离散小波变换就是用多分辨率的形式分解信号, 消除信号间的相关性. 在每层分解信号都是用小波信号分解为高频段信号和低频段信号, 即分别由相应的高通滤波器和低通滤波器来获得. 提升方法是实现这些滤波运算的一个有效方法, 而且方法相当简单. 它使用基本的多项式插补来获取信号的高频分量, 然后通过构建尺度函数来获取信号的低频分量. 每一步提升包含三个步骤分解, 预测和更新. 如图 11所示.

分解就是把信号 a_{t-1} , 分为偶数抽样点 a_t , 和奇数抽样点 d_t , 这也称为懒小波变换 (Lazy Wavelet Transform). 设信号 a_{t-1} 为有限长度的一维离散输入序列, 它们之间存在一定的相关性. 为了消除相关性, 首选把信号 a_{t-1} 分解为两个子信号 a_t 和 d_t . 一般来说, 对于信号分解的形式、两个子信号的大小相等, 没有什么特别的限制, 但要求存在某一与分解相对应的算子, 可以从子信号 a_t 和 d_t , 还原到 a_{t-1} , 这是离散小波变换和子带编码中最基本的要求, 是由完全重构性质所决定的. 在离散小波变换和子带编码中常用的是二通道向下抽样 (Downsampling), 也就是把信号 a_{t-1} 分解为偶数采样点 a_t , 和奇数抽样点 d_t . 这个变换实际上什么也没做, 对信号表示形式也没什么改进, 但这是后面步骤的基础, 所有的二进小波变换都是从这一步开始的.

预测也被称对偶提升 (Dual Lifting), 就是由 a_t 预测 d_t , 用预测误差代替 d_t , a_{t-1} 之间存在一定的相关性, 故可以从 a_t 估计 d_t , 令 $d_t = P(a_t)$ 这就是预测. 也可以说, 是将 a_t 作为 a_{t-1} 的近似值. 若信号之间的相关性很大, 那么预测效果会很好, 将 a_t 作为 a_{t-1} 的近似表示不会“丢失”很多信息. 这就意味着可以“扔掉”部分信息, 即 d_t , 以达到简练表示的目的. 为了完全重建信号 a_{t-1} , 就只能“扔掉”包含在 d_t 中的关于 a_t 的那部分信息, 即可用 $d_t - P(a_t)$ 代替 d_t , 既达到消除相关性的目的, 也能保证存在某种逆算子, 可以重建信号 a_{t-1} .

更新又称为主要提升 (Primal Lifting), 即用 d_t 更新 a_t , 预测后得到的这样一种新的表示形式中, 很可能会丢失信号的某些特征, 如信号的均值, 而这正是我们所期望的如对信号进行压缩时. 为了恢复这些特征, 在提升算法中又引入了另外一种操作——更新 (U), 即用新得到的 d_t 来更新 a_t .

如图 12所示, 提升方法可以实现原位运算, 即该算法不需要除了前级提升步骤的输出之外的数据, 这样在每个点都可以用新的数据流替换旧的数据流. 当重复使用原位提升滤波器组时, 就获得了交织的小波

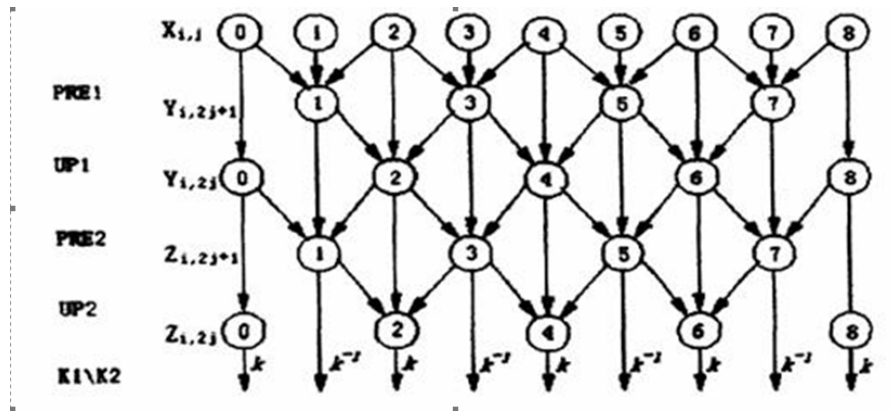


图 12: 提升算法的相关性

变换系数. 由于小波变换需要较大的计算量, 且计算复杂度高, 靠软件实现无法满足实用需要, 其硬件实现日益受到重视. 其中, 基于 FPGA 的小波变换及编码方法一直是研究的热点.

7 参考文献

1. <https://blog.csdn.net/u012654981/article/details/81984885>
2. <https://github.com/mohitkr/wavelets>
3. https://github.com/jswanglp/NN_and_WNN
4. <https://wenku.baidu.com/view/b582d50c1611cc7931b765ce0508763230127458.html>
5. Olkkonen J T , Olkkonen H . Discrete Lattice Wavelet Transform[J]. IEEE Transactions on Circuits and Systems II: Express Briefs, 2007, 54(1):71-75.
6. 余纯武. $L^2(-1, 1)$ 空间的多项式小波分解 [J]. 湖北大学学报: 自然科学版, 1998, 20(1):15-18.
7. 张晓慧. 小波调制调制度的引入及其分析 [J]. 自动化技术与应用, 2017(10).