Phase unwrapping using discrete Particle Swarm Optimization

Wei He[1], Ling Xia[1], Yiyuan Cheng[1], Feng Liu[2], and Stuart Crozier[2]

[1]Department of Biomedical Engineering, Zhejiang University, Hangzhou, Zhejiang, China, People's Republic of, [2]The School of Information Technology and Electrical Engineering, The University of Queensland, Brisbane, Australia

Introduction: The phase of the MRI signal can provide important information about the magnetic field inhomogeneities, magnetic susceptibility variations, velocity of flowing spins and so on. However, when extracting the phase $\psi(x,y)$ from a measured complex MRI image $I(x,y)=|I(x,y)|exp(\psi(x,y))$, through some mathematical operation, the result is typically wrapped into the principal interval of $(-\pi, \pi]$, producing the wrapped phase $\varphi(x,y)$. The process of estimating the true phase from the wrapped phase is called phase unwrapping. Because of the presence of the noise, undersampling and/or object discontinuities, phase unwrapping becomes difficult. One class of phase unwrapping algorithm is called branch-cut phase unwrapping. It detects residues, connects the residues of opposite polarity by branch cuts and then unwraps the phase to avoid those branch cuts. This approach tries to minimize the total length of the branch cuts; hence, it will decrease the amount of good pixels in the branch cuts and obtain a smoother result. We propose a new discrete Particle Swarm Optimization (dPSO)-based[1] branch-cut phase-unwrapping algorithm. Two novel features of the proposed dPSO method are, the grouping of all of the residues by dividing the phase image into some sub-regions, and in each region using the dPSO to match the two opposite polarity residues in pairs for the purpose of minimizing the overall length of the cuts and the area of the isolated patches.

Methods: Residues division. We divide the whole phase image into sub-regions according to its phase derivative variance map, defined by the equation: $z_{m,n} = [\sqrt{\sum (\Delta\varphi^{x}_{i,j} - \overline{\Delta\varphi^{x}_{m,n}})^2} + \sqrt{\sum (\Delta\varphi^{y}_{i,j} - \overline{\Delta\varphi^{y}_{m,n}})^2}\,] / l^2$ (Eq.(1)), where for each sum, the indexes (i,j) range over the $l\times l$ window centered at the pixel (m,n). The terms $\Delta\varphi^{x}_{i,j}$ and $\Delta\varphi^{y}_{i,j}$ are the wrapped-phase differences, $\overline{\Delta\varphi^{x}_{m,n}}$ and $\overline{\Delta\varphi^{y}_{m,n}}$ are the averages of these wrapped-phase differences in $l\times l$ windows. In this paper, $l = 3$. Because most of the residues distribute in the regions of high phase derivative variance values, based on a suitable threshold, we convert the phase derivative variance map to a binary map and the residues almost scatter in the pixels of value 1. Then, these pixels are inflated to shape the separate areas. In this way, the residues are grouped.

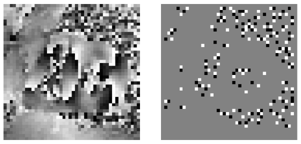

Figure1. A 44×44 MRI wrapped-phase image (Left), and its residue distribution (Right).

The dPSO algorithm. Initially, the indexes of the residues in each region are inserted in two arrays, positive residue and negative residue arrays. The positive array will be fixed throughout all iterations and acts as a reference. However, the negative array will be used as a particle. That is, searching the best match order of the negative residues is the target. If there are N negative residues in the current region, the *i*th particle of swam is represented as $U_i=\{u_{i1}, u_{i2},..., u_{iN}\}$, where each element is an index of a negative residue. The local best position, the *i*th particle has reached, is recorded as $P_i=\{p_{i1},p_{i2},\ldots,p_{iN}\}$. The global best position is $P_g=\{p_{g1},p_{g2},\ldots,p_{gN}\}$. The velocity of the *i*th particle is represented as $V_i=\{v_{i1},v_{i2},\ldots,v_{iN}\}$. In each iteration, every particle changes its velocity and position by the following formulas: $V_i^{t+1} = w * V_i^t + c_1 * rand() * (P_i^t - U_i^t) + c_2 * Rand() * (P_g^t - U_i^t)$ (Eq.(2)), $U_i^{t+1} = U_i^t + V_i^{t+1}$ (Eq.(3)), where t represents the iteration number, c_1 and c_2 are non-negative constants, called learning factors, determining the influence of P_i and P_g, the function rand() and Rand() generate a random number between 0 and 1, w is the inertia weight. Considering the capacity of searching the global optimal solution and convergence rate, we define w to linearly decrease during all the iterations: $w = 0.9 - 0.5 \times (t / T)$ (Eq.(4)), where T is the total iteration times. Then, we define a permutation operator $PO(a_1,a_2)$ for swapping the a_1th *and* a_2th indexes in a particle. The plus sign '+' between a particle and a permutation operator means acting the operator on the particle, while '+' between permutation operators or permutation operator arrays means putting the latter behind the former. The order of the operators in the array is important. Thus, the operators cannot be interchanged and must be implemented in turn. The minus sign '-' means constructing an array of one or more permutation operators that can apply to the item behind the sign to get the one ahead. Therefore, the velocity V_i is an array of permutation operators. In addition, the multiplication sign '*' between a real number and the permutation operator array means reserving a part of the array when the real number is in the range (0,1). The fitness function is shown as: $\text{Fitness} = \sum_j \sqrt{(x_j^+ - x_{u_{ij}})^2 + (y_j^+ - y_{u_{ij}})^2}$ (Eq.(5)).

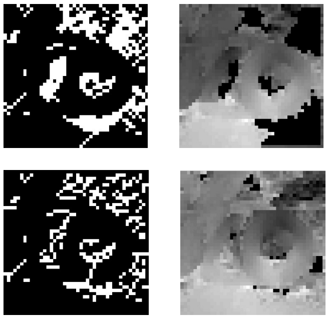

Figure 2. The branch-cut distribution (Left) and unwrapped-phase image (Right), achieved by using the Goldstein's algorithm (Upper) and the dPSO algorithm (Lower).

Branch cuts and unwrapping. Once the best match in every region is found by the dPSO, the branch cuts are placed to connect each pair. The nearest neighbor algorithm [2] is employed to place branch cuts among any residues left. Then the phase can be unwrapped by the flood-fill algorithm [3].

Results: A MRI dataset [4] was used to verify the performance of the proposed algorithm. The results were compared with Goldstein's algorithm [3]. The MRI wrapped-phase image and its corresponding residue distribution are shown in Figure1. The corresponding branch-cut distribution and unwrapped-phase image by using Goldstein's method are shown in Figure 2 (Upper). It is easy to determine that there are two large patches isolated in the upper part of the unwrapped image, two small ones in the middle and a very large one in the lower right part. The dPSO results shown in Figure 2 (Lower) are achieved by using 0.9612 as the threshold, a swarm of 300 particles, *c1*=2, c2=2 and *T*=1000. The comparison is shown in Table 1.

Table 1. Comparison of dPSO with Goldstein's algorithms by total cuts length, unweighted L^0 measure and area of the isolated patches.

Algorithm	Total Cuts Length	Unweighted L^0 Measure	Area of the Isolated Patches
dPSO	309	0.2283	41
Goldstein's	454	0.3755	382

Conclusion: A new dPSO-based branch-cut phase unwrapping algorithm has been proposed. This algorithm has been demonstrated to be robust and effective. The result of the dPSO was compared to Goldstein's algorithm. It is deduced that the dPSO is a better algorithm in terms of the total branch cuts length, unweighted L^0 measure and the area of the isolated patches. It is important to point out that the complexity of the dPSO algorithm increases with the increase in the number of residues. This is attributable to the increase in the length of the particle, which will require a larger swarm size. Future modifications of this algorithm will enable it to be faster and set the appropriate threshold by itself.

References: [1] K. P. Wang, et al. *Proceedings of the International Conference on Machine Learning and Cybernetics*. 1583 – 1585 (2003). [2] R. Cusack, J. M. Huntley and H. T. Goldrein. *Appl. Opt.* 34, 781-789 (1995). [3] D. C. Ghiglia and M. D. Pritt, *John Wiley & Sons, INC.* New York. 100-310 (1998). [4] B. Spottiswoode. http://www.mathworks.com/matlabcentral/fileexchange/22504.