

# HaraliCU: GPU-Powered Haralick Feature Extraction on Medical Images Exploiting the Full Dynamics of Gray-scale Levels

Leonardo Rundo<sup>\*1,2,3</sup>, Andrea Tangherloni<sup>\*4,5,6,3</sup>, Simone Galimberti<sup>3</sup>,  
Paolo Cazzaniga<sup>7,8</sup>, Ramona Woitek<sup>1,2,9</sup>, Evis Sala<sup>1,2</sup>,  
Marco S. Nobile<sup>3,8</sup>, and Giancarlo Mauri<sup>3,8</sup> (✉)

<sup>1</sup> Department of Radiology, University of Cambridge,  
Cambridge, United Kingdom

<sup>2</sup> Cancer Research UK Cambridge Centre, Cambridge, United Kingdom

<sup>3</sup> Department of Informatics, Systems and Communication,  
University of Milano-Bicocca, Milan, Italy  
`giancarlo.mauri@unimib.it`

<sup>4</sup> Department of Haematology, University of Cambridge,  
Cambridge, United Kingdom

<sup>5</sup> Wellcome Trust Sanger Institute,  
Wellcome Trust Genome Campus, Hinxton, United Kingdom

<sup>6</sup> Wellcome Trust – Medical Research Council Cambridge  
Stem Cell Institute, Cambridge, United Kingdom

<sup>7</sup> Department of Human and Social Sciences, University of Bergamo,  
Bergamo, Italy

<sup>8</sup> SYSBIO.IT Centre of Systems Biology, Milano, Italy

<sup>9</sup> Department of Biomedical Imaging and Image-guided Therapy,  
Medical University Vienna, Vienna, Austria

**Abstract.** Image texture extraction and analysis are fundamental steps in Computer Vision. In particular, considering the biomedical field, quantitative imaging methods are increasingly gaining importance since they convey scientifically and clinically relevant information for prediction, prognosis, and treatment response assessment. In this context, radiomic approaches are fostering large-scale studies that can have a significant impact in the clinical practice. In this work, we focus on Haralick features, the most common and clinically relevant descriptors. These features are based on the Gray-Level Co-occurrence Matrix (GLCM), whose computation is considerably intensive on images characterized by a high bit-depth (e.g., 16 bits), as in the case of medical images that convey detailed visual information. We propose here HaraliCU, an efficient strategy for the computation of the GLCM and the extraction of an exhaustive set of the Haralick features. HaraliCU was conceived to exploit the parallel computation capabilities of modern Graphics Processing Units (GPUs), allowing us to achieve up to  $\sim 20\times$  speed-up with respect to the corresponding C++ coded sequential version. Our GPU-powered solution highlights the promising capabilities of GPUs in the clinical research.

---

\* These authors contributed equally.

**Keywords:** Haralick features · GPU computing · Full gray-scale range · Medical imaging · Radiomics · CUDA

## 1 Introduction

Texture analysis has been effectively used in the classification and categorization of pictorial data in several Computer Vision tasks, such as object detection [1] and representation [2]. More specifically, texture features allow for quantitative analyses of the properties concerning scenes or objects of interest. Even though Deep Learning has recently gained ground, conventional Machine Learning models built on top of hand-engineered features remain fundamental in practical applications, especially thanks to the interpretability of the results [3]. With particular reference to biomedicine, quantitative imaging methods are increasingly gaining importance since they convey scientifically and clinically relevant information for prediction, prognosis, and treatment response assessment [4]. In this context, radiomic approaches are encouraging large-scale studies that can have a significant impact in the clinical practice [5]. Radiomics aims at extracting huge amounts of features from medical images and then mining them by means of cutting-edge computational techniques [6]. By so doing, radiomics exploits advanced imaging features to objectively and quantitatively describe tumor phenotypes [5]. Recently, radiomic studies have drawn considerable interest due to the potentialities for predicting treatment outcomes and cancer genetics, which may have important applications in personalized medicine [6, 7]. Relying on the idea that radiomic features convey information about the different cancer phenotypes, they enable quantitative measurements for intra- and inter-tumoral heterogeneity.

The radiomic features can be essentially divided into four classes [8]. The first class comprises features related to region-based measurements (i.e., size, shape, diameter), while the other classes can be described as first-, second-, and higher-order statistical outputs, respectively. First-order statistical features concern the gray-level intensity histogram of a Region of Interest (ROI), such as mean, median, standard deviation, minimum, maximum, quartiles, kurtosis, and skewness. The second-order statistics consider texture analysis, which describes the texture of the ROI, by relying on the Gray-Level Co-occurrence Matrix (GLCM) that stores the co-occurrence frequency of similar intensity levels over the region (i.e., intensity value pairs). An alternative technique belonging to the second-order statistical outputs is fractal-based texture analysis, which examines the difference between pixels at different length scales (i.e., offset differences) [9]. Lastly, the higher-order methods extract repetitive or non-repetitive patterns by using kernel functional transformations, as in the case of the Gray-Level Run Length Matrix (GLRLM), which gives the size of homogeneous runs for each gray-level [10], and the Gray-Level Zone Length Matrix (GLZLM), which provides information on the size of homogeneous zones for each gray-level [11]. Moreover, some popularly used descriptors in transformed domains are Fourier transform, Wavelets, and Gabor filters [12, 13].

Among the available radiomic descriptors, Haralick features are the most commonly used and clinically relevant [14, 15], allowing radiologists to assess image regions characterized by heterogeneous/homogeneous areas or local intensity variations [16]. GLCM-based texture features have been extensively exploited in several medical image analysis tasks, such as breast Ultrasound (US) classification [17], brain tissue segmentation on Magnetic Resonance (MR) images [18], and volume-preserving non-rigid lung Computed Tomography (CT) image registration [19]. Unfortunately, the computation of these features is considerably intensive on images characterized by a high bit-depth (e.g., 16 bits), such as in the case of medical images that have to convey detailed visual information [20]. As a matter of fact, with the existing computational tools, the range of intensity values of an image must be reduced and limited to achieve an efficient radiomic feature computation [7].

In this work, we propose a novel strategy to compute the GLCM and extract an exhaustive set of the Haralick features. In particular, we aim at overcoming the limitations of the available feature extraction and radiomics tools that cannot effectively manage the full-dynamics of gray-scale levels. Our method, called HaraliCU, can offload the computations onto the GPU cores, thus allowing us to drastically reduce the running time required by the execution on Central Processing Units (CPUs).

This manuscript is organized as follows. Sect. 2 introduces the fundamental concepts regarding the GLCM-based textural features by presenting also the set of the extracted Haralick features. Sect. 3 introduces the Compute Unified Device Architecture (CUDA) and summarizes the state-of-the-art of the available software for Haralick feature extraction. Sect. 4 describes HaraliCU in details. The achieved results are shown and discussed in Sect. 5. Finally, some concluding remarks and future developments of this work are given in Sect. 6.

## 2 Haralick Features

Haralick features contain data about image textural characteristics, e.g., homogeneity, gray-tone linear dependencies, contrast, number and nature of boundaries present, along with indices of the inherent complexity of the image. All these features are calculated according to a GLCM.

### 2.1 GLCM: Basic Concepts

Formally, a GLCM with size  $L \times L$ , where  $L$  represents the maximum number of gray-levels according to the quantization scheme, denotes the second-order joint probability function  $p(i, j)$  of an image region—where  $i, j \in [0, 1, \dots, L - 1]$  are gray-levels—defined as  $\mathbf{P}(i, j)$ . The GLCM considers the mapping of the initial full dynamics due to computational limitations.

In what follows, we will refer to two neighboring pixels, separated by a distance  $\delta$  along an orientation  $\theta$ , as the pair  $\langle \text{reference, neighbor} \rangle$ , where the **reference** pixel has gray-level equal to  $i$ , while the **neighbor** pixel is characterized by a gray-level

$j$ . More specifically, given a sliding window of size  $\omega \times \omega$ , the  $\langle i, j \rangle$ -th element of the matrix  $\mathbf{P}(i, j | \delta, \theta, \omega)$  represents the number of times that the combination of the levels  $i$  and  $j$  occurs in two pixels (reference, neighbor) inside the sliding window, which are separated by a distance of  $\delta$  pixels along the orientation  $\theta$ . The distance  $\delta$  is defined according to the infinity norm  $\ell_\infty$ . The undirected and directed distances denote the symmetric and non-symmetric GLCM, respectively, in terms of conditional co-occurrence probabilities. In some specific applications, valuable information could be lost in the symmetric approach [15]. Specifically:

- when computing the symmetric GLCM  $\mathbf{P}_s$ , since the pairs of gray-levels  $\langle i, j \rangle$  and  $\langle j, i \rangle$  are considered as the same element in  $\mathbf{P}_s$ , the frequency of both  $\langle i, j \rangle$  and  $\langle j, i \rangle$  is increased, so the resulting GLCM is symmetric across its main diagonal;
- when computing the non-symmetric GLCM  $\mathbf{P}_{ns}$ , the pairs of gray-levels  $\langle i, j \rangle$  and  $\langle j, i \rangle$  are considered separately in  $\mathbf{P}_{ns}$ .

In medical imaging, the selection of  $\delta$  and  $\theta$  used for the GLCM computation could depend on the specific application. For instance in breast US, the direction  $\theta = 90^\circ$  coincides with the direction of US propagation [17]. In order to obtain rotationally invariant features, it is common to average the GLCM-based statistics achieved over the four directions  $\theta \in \{0^\circ, 45^\circ, 90^\circ, 135^\circ\}$ .

## 2.2 Haralick Features in Medical Imaging

As a first step, we conducted an in-depth analysis of the literature to accurately define an exhaustive set of the Haralick features and avoid both ambiguities and redundancies. In the literature, some features exhibited potential in the characterization of the cancer imaging phenotype. For instance, entropy was shown to be a promising quantitative imaging biomarker for characterizing cancer heterogeneity, although it could be affected by acquisition protocols in multi-institutional studies [21]. With regard to the computation of the GLCM-based features, HaraliCU exploits the existing dependencies among Haralick features. Indeed, Gipp *et al.* [22] pointed out that some features can exploit some calculations pertaining to other features or intermediate results.

Considering the process of image digitalization, the compression of the initial intensity range is called quantization, which is generally irreversible and results in loss of information. For instance, in the case of texture features based on the Standardized Uptake Value (SUV) [23] within the tumor, a quantization phase is involved. Orlhac *et al.* [24] compared the different quantization strategies in metabolic activity pattern identification, by showing that they might significantly affect the texture values. In [25], the Positron Emission Tomography (PET)-derived texture features were calculated by quantizing the tumor voxel intensities with similar uptake to the same value. A similar study on Haralick features computed on the Apparent Diffusion Coefficient (ADC) MR images was presented in [16]. Even though the authors claimed that the impact of noise is reduced, this gray-scale compression could considerably decrease the discriminating power

in feature-based classification tasks [26]. However, the main practical argument for the gray-scale compression is the computational cost. Therefore, to fully justify this choice, more advanced and adaptive quantization schemes should be devised [16]. With reference to the normalization of CT-based radiomics [27], the influence of gray-level quantification on radiomic feature stability for different CT scanners, tube currents and slice thickness was investigated in [28].

### 3 State-of-the-art

The main limitation of the existing radiomics tools concerns their inability to deal with the full dynamics of 16-bit images, meaning that they are not capable of extracting the feature maps by preserving the initial gray-scale range. This drawback is emphasized when handling feature extraction tasks on the whole input image, especially for image classification purposes.

The aim of our approach is to tackle these issues, by effectively computing the feature maps for high-resolution images with their full dynamics. Since the calculation of Haralick features represents an embarrassingly parallel problem, several High-Performance Computing (HPC) technologies can be exploited. Among them, General-Purpose Computing on GPUs (GPGPU) is one of the most promising approaches. With reference to the existing Single Instruction Multiple Data (SIMD) architectures, NVIDIA CUDA is one of the most widespread and popular options [29]. CUDA is designed to exploit the parallelism provided by many-core GPUs for general-purpose scientific computing. Specifically, the idea is to offload intrinsically parallel calculations from the CPU, called the host, onto one or more devices (the GPUs) by means of kernels, that is, functions launched from the host and replicated in multiple threads running on the GPU cores.

CUDA threads are logically subdivided into thread blocks which, in turn, are organized in block grids. From a hardware standpoint, blocks are distributed over the GPU Streaming Multiprocessors (SMs) for their execution. When the blocks outnumber the available SMs, they are queued by the CUDA scheduler, transparently scaling the performance on different GPUs. Indeed, the higher the number of SMs, the higher the number of blocks running at the same time. The threads in execution on an SM are organized in tight groups of 32 threads named warps, which are executed in locksteps. Thus, blocks smaller than 32 threads imply a reduced occupancy of the GPU resources. In addition, due to this peculiar pattern of execution, any divergent path taken by some threads in a warp (e.g., the consequence of a conditional `if-then-else` statement) causes a serialization of the execution until re-convergence, affecting the overall performance. Therefore, in order to achieve optimal performance, CUDA code must be optimized to prevent any branch divergence in the execution.

CUDA has also a complex memory hierarchy, characterized by multiple memory types that provide different advantages and drawbacks. Notable examples are the global memory (large, visible by all threads, and affected by high access latency) and the shared memory (very small and used for intra-block communications with very slow access latency). A careful optimization of the data structures

in these memories is mandatory to achieve the theoretical peak performance. Moreover, since any memory transfer between the host and device is very time consuming, they should be reduced as much as possible. Due to these peculiarities, CUDA programming could be challenging and generally requires the redesign of existing algorithms [30].

GPUs are representing an enabling factor for feasible computational solutions in medical image analysis [31, 32]. Over the last years, GPUs proved to be fundamental for the practical use of computationally demanding algorithms [30], like the efficient training of deep neural networks [33]. Considering the GPU-accelerated Haralick feature extraction methods, Gipp *et al.* [22] proposed a packed representation of the symmetric GLCM, by storing only the rows and columns with non-zero elements. Afterwards, the Haralick features are computed by means of the lookup table that maps the index of the packed co-matrix. This clever solution reduces the accesses to global memory and, in turn, reduces the latencies due to memory reads, strongly improving the overall performances. The authors applied this implementation to cell images with 12-bit intensity depth. Tsai *et al.* [34] proposed an indirect encoding scheme for storing the GLCM, named the meta GLCM array, designed to fully exploit the GPU memory hierarchy. This approach was tested on brain MR images.

## 4 The Proposed GPU-accelerated Method

Medical images convey a valuable amount of information, in terms of image resolution as well as pixel depth, which should be maintained for automated processing [20], since additional clinically useful pictorial details could be identified with respect to the naked eye perception. For these motivations, in the proposed approach, we aimed at keeping the whole initial information provided by the full dynamics of the gray-levels (i.e., 16 bits in the case of biomedical images), by efficiently managing the memory. As a matter of fact, the state-of-the-art methods exhaust the physical memory, such as in the case of the MatLab built-in function `graycomatrix`, even if running on machines equipped with 16 GB of RAM.

HaraliCU aims at supporting the user by providing low-level control. Indeed, the user can set the distance offset  $\delta$ , the orientation  $\theta$ , and the window size  $\omega \times \omega$ , while the neighborhood  $\mathcal{N}$  is defined according to  $\delta$  and  $\theta$ . Therefore, the features can be computed for the four directions and then averaged to obtain a single aggregate value. The user can also set the padding conditions for the border pixels, either by choosing the zero padding or the symmetric padding. The number of quantized gray-levels  $Q$  can be also provided; HaraliCU linearly maps the initial minimum and maximum gray-levels onto 0 and  $Q - 1$ , respectively, in order to avoid the loss of a considerable amount of intensity bins.

The accuracy of the proposed efficient GLCM computation approach was evaluated against the built-in function `graycomatrix` provided by MatLab. The computation of the Haralick features was carefully compared against the `graycoprops` function, which provides only the contrast, correlation, energy (i.e.,

angular second moment), and homogeneity features. For the other features, we relied also on a MatLab implementation publicly available on MatLab Central<sup>1</sup>. It is worth to note that our comparison was limited to the use of  $L = 2^8$  gray-levels for the computation of the GLCM due to the computational limitations of the MatLab implementation. As a matter of fact, the `graycomatrix` function requires a double-precision  $L \times L$  GLCM, by exceeding the main memory even in the case of 16 GB of RAM.

Since allocating a GLCM with  $2^{16}$  rows and columns for each sliding window is memory demanding, and also considering that the size of each GLCM is strictly related to the number of different gray-levels inside the considered sliding window, we designed an effective and efficient encoding. More specifically, our novel encoding consists in storing each GLCM by using a list-based data structure in which every element of the list is a pair  $\langle \text{GrayPair}, \text{freq} \rangle$ , where `GrayPair` is a pair  $\langle i, j \rangle$  of gray-levels and `freq` is the corresponding frequency (i.e., number of occurrences of the pair  $\langle i, j \rangle$ ) inside the considered sliding window. The number of possible different elements composing the GLCM is given by the number of pairs  $\langle \text{reference}, \text{neighbor} \rangle$  that can be identified inside the sliding window, taking into account the distance  $\delta$ . The exact number of elements is provided by the following equation:  $\#\text{GrayPairs} = \omega^2 - \omega\delta$ . The GLCM is dynamically computed by using the following procedure:

1. each pair  $\langle \text{reference}, \text{neighbor} \rangle$ , with gray-levels equal to  $\langle i, j \rangle$ , belonging to the sliding window is evaluated;
2. when a pair  $\langle i, j \rangle$  is found, if the corresponding `GrayPair` element in the list exists, its frequency `freq` is incremented; otherwise, a new element `GrayPair`, with `freq` equal to 1, is allocated and appended to the end of the list.

This simple but efficient encoding allows for removing all the zero elements inside the GLCM. In addition, when the GLCM symmetry is exploited, the length of the list is halved: indeed, the pairs  $\langle i, j \rangle$  and  $\langle j, i \rangle$  are considered as the same pair and the frequency of the pair  $\langle i, j \rangle$  is doubled.

Considering that there are no dependencies between the sliding windows, we assigned each pixel of the input image to a GPU thread. In such a way, each thread computes all the features related to its pixel, which represents the center of the corresponding window. As a matter of fact, since a medical image could be often composed of more than 250 thousand pixels, involving the same number of sliding windows in the feature map computation, GPUs—thanks to their high number of threads that can be executed in parallel—are the most suitable co-processors to parallelize the required massive computational workload. In order to maximize the GPU performance and to fully exploit the GPU acceleration, we created a bi-dimensional structure for both the number of blocks and the number of threads. We fixed the number of threads to 16 for both the components of the bi-dimensional structure, while the number of blocks for each component of the corresponding bi-dimensional structure strictly depends on the number of the

<sup>1</sup> <https://uk.mathworks.com/matlabcentral/fileexchange/22187-glcm-texture-features>

pixels (`#pixels`) composing the input image, and can be calculated as follows:

$$n_{\text{blocks}} = \begin{cases} \hat{n}, & \text{if } \hat{n}^2 \geq \lceil \frac{\text{\#pixels}}{256} \rceil \\ 1, & \text{otherwise} \end{cases} .$$

We used 16 threads in each component to take into consideration the CUDA warp size (i.e., 32 threads) as well as the limited number of registers.

Each thread processes a sliding window, that is, a subset of the pixels of the original image. Hence, all threads fetch from the GPU’s global memory the pixels that are necessary for the calculations. However, some pixels may be shared by partially overlapping windows, a circumstance that introduces unnecessary latencies in the execution and might be mitigated by exploiting the shared memory. We will investigate this feature in a next release of HaraliCU.

## 5 Experimental Results

As described in the previous section, we validated HaraliCU by comparing the values of the features contrast, correlation, energy, and homogeneity with those extracted using the built-in functions `graycomatrix`.

### 5.1 Test Images

For the tests presented here, we considered two medical datasets characterized by different modalities and image size:

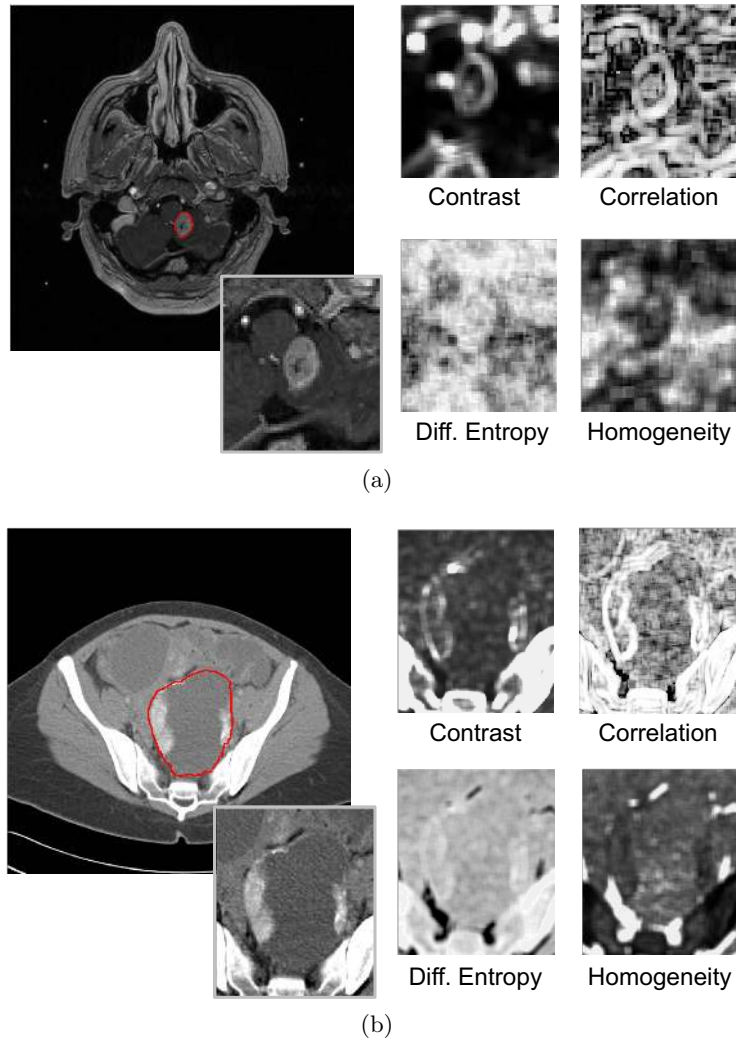
- axial T1-weighted Fast Field Echo contrast-enhanced MR sequences of brain metastases (matrix size:  $256 \times 256$  pixels, pixel spacing: 1.0 mm, slice thickness: 1.5 mm), where the extracted features can be applied to segmentation and classification tasks [18, 35];
- axial contrast-enhanced CT series of high-grade serous ovarian cancer (matrix size:  $512 \times 512$  pixels, pixel spacing:  $\sim 0.65$  mm, slice thickness: 5.0 mm), where texture features can evaluate intra- and inter-tumoral heterogeneity [36, 37]. Pelvic lesions only were selected for this work.

In both cases, the intensity depth is 16 bits.

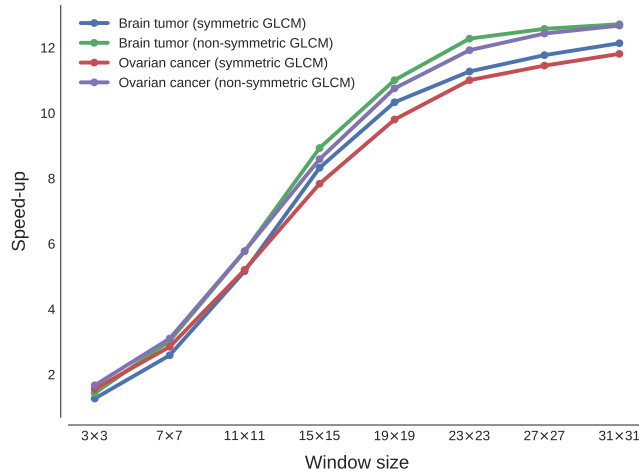
### 5.2 Computational Results

The existing versions of Haralick feature extraction tools are typically characterized by prohibitive running times, making them unfeasible in the clinical research. Moreover, these tools are not capable of taking into consideration the full dynamics of gray-scale levels; we therefore developed a memory-efficient CPU version of HaraliCU (coded in C++), which overcomes this limitation and was also used as a benchmark to show the advantages of exploiting the GPUs to accelerate the calculations required by this computationally intensive task.





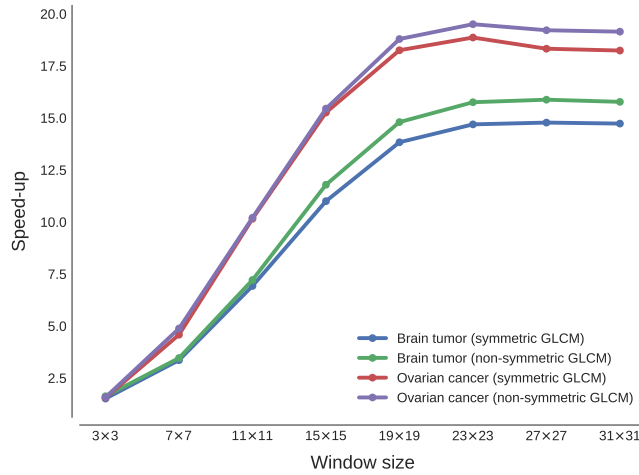
**Fig. 1.** Examples of feature maps obtained by HaraliCU by considering the full dynamics of gray-scale levels: (a) axial contrast-enhanced T1-weighted MR image of enhancing brain metastatic cancer; (b) axial venous phase contrast enhanced CT image of a patient with high-grade serous ovarian cancer showing the partly calcified and cystic ovarian tumor (red ROI) and omental disease (not outlined). The original images are shown in the leftmost panel. The ROIs (i.e., the tumor regions) are highlighted with a red contour and the corresponding cropped sub-images containing the ROIs are zoomed at the bottom right of each sub-figure. In the rightmost panel, we show four selected feature maps for the ROI-centered cropped images, namely: contrast, correlation, difference entropy and homogeneity. In both cases, the features were extracted by using  $\delta = 1$  and averaging over  $\theta \in \{0^\circ, 45^\circ, 90^\circ, 135^\circ\}$  to enrich the visual content. We selected  $\omega = 5$  and  $\omega = 9$  for the brain metastasis MR and the ovarian cancer CT images, respectively.



**Fig. 2.** Speed-up achieved by the GPU-powered version of HaraliCU, with respect to the C++ counterpart, on brain metastatic tumor MR and ovarian cancer CT images, by considering  $2^8$  intensity levels, enabling and disabling the GLCM symmetry, and considering  $\omega \in \{3, 7, 11, 15, 19, 23, 27, 31\}$ . Blue and green lines denote the speed-up trend considering brain metastasis MRI images, while red and violet lines are used for ovarian cancer CT images.

We first show in Figs. 1a and 1b two examples of input images along with the corresponding feature maps of four selected descriptors in the case of brain metastatic tumor MR and ovarian cancer CT images, respectively, to evaluate the correctness of our implementation. From the computational point of view, our C++ implementation resulted extremely efficient with respect to the MatLab version, based on the `graycomatrix` and `graycoprops` functions, to extract Haralick features on a brain metastasis MR image. As a matter of fact, by varying the gray-scale range from  $2^4$  to  $2^9$  levels, we achieved speed-up values around  $50\times$  and  $200\times$ , respectively.

As a second step, we compared the computational performance of our single core CPU version and GPU-powered versions of HaraliCU to extract all the provided features, to assess the capabilities of the parallel implementation. The GPU version of HaraliCU was run on an NVIDIA GeForce GTX Titan X (3072 cores, clock 1.075 GHz, 12 GB of RAM), CUDA toolkit version 8 (driver 387.26), running on a workstation with Ubuntu 16.04 LTS, equipped with a CPU Intel Core i7-2600 CPU (clock 3.4 GHz) and 8 GB of RAM. The CPU version of HaraliCU was run on the same workstation, relying on the computational power provided by the CPU Intel Core i7-2600 CPU. The CPU version was compiled by using the GNU C++ compiler (version 5.4.0) with optimization flag `-O3`, while the GPU version was compiled with the CUDA Toolkit 8.0 by exploiting the optimization flag `-O3` for both CPU and GPU code.



**Fig. 3.** Speed-up achieved by the GPU-powered version of HaraliCU, with respect to the C++ counterpart, on brain metastatic tumor MR and ovarian cancer CT images, by considering  $2^{16}$  intensity levels, enabling and disabling the GLCM symmetry, and considering  $\omega \in \{3, 7, 11, 15, 19, 23, 27, 31\}$ . Blue and green lines denote the trend considering brain metastatic tumor MRI images, while red and violet lines are used for ovarian cancer CT images.

In order to collect statistically sound results and take into consideration the variability and the heterogeneity typically characterizing these images, we randomly selected 30 images from 3 different patients (10 per patient) affected by brain metastases and 30 images from 3 different patients affected by ovarian cancer. We tested both versions of HaraliCU by considering different window sizes, that is,  $\omega \in \{3, 7, 11, 15, 19, 23, 27, 31\}$ , as well as two different intensity levels (i.e.,  $2^8$  and  $2^{16}$ ). For each combination of  $\omega$  and intensity levels, we also enabled and disabled the GLCM symmetry to evaluate how the symmetry affects the running time of HaraliCU. It is worth noting that the measurements of the execution time of HaraliCU include the data transfer between the host memory and the device memory.

Figs. 2 and 3 show the speed-up achieved by the GPU-powered version of HaraliCU. Considering only  $2^8$  intensity levels, the speed-up increases almost linearly; in addition, by disabling the GLCM symmetry and using  $\omega = 31$  we obtained the highest speed-ups of  $12.74\times$  and  $12.71\times$  on brain metastasis ( $256 \times 256$  pixels) and ovarian cancer images ( $512 \times 512$  pixels), respectively. When the full dynamic of the gray-scale levels (i.e.,  $2^{16}$ ) is considered, the GPU-powered version of HaraliCU outperforms the sequential counterpart, achieving speed-ups up to  $15.80\times$  with  $\omega = 31$  and  $19.50\times$  with  $\omega = 23$ , on brain metastasis and ovarian cancer images, respectively. Taking into account ovarian cancer images, when  $\omega$  is greater than 23 pixels the speed-up decreases for two reasons. Firstly, each thread, which is associated with a pixel, must consider more neighbor pixels

that might have very different gray-level intensities, since their values are in the range  $[0, 1, \dots, 2^{16} - 1]$ . This corresponds in increasing the required workload that each thread must perform; since the GPU cores have a lower clock frequency than CPU cores, the speed-up is reduced. Secondly, the GPU resources are saturated since the GLCM size associated with each thread may increase due to the high full-dynamic range. In this specific situation, the total GLCM size might overwhelm the dimension of the global memory and some threads handle different pixels, computing the corresponding Haralick features in a sequential way, decreasing the number of threads running in parallel.

The source code and the instructions for the compilation and execution of HaraliCU are available under the GNU GPL v3.0 license on GitHub at the following URL: <https://github.com/andrea-tango/HaraliCU>. HaraliCU requires an NVIDIA GPU, CUDA toolkit version 8 (or higher), OpenCV library version 3.4.1 (or higher).

## 6 Conclusion

Image texture extraction and analysis is playing a key role in quantitative biomedicine, leading to valuable applications in radiomic [5, 6] and radiogenomic [38] research, by also combining heterogeneous sources of information. Therefore, advanced computerized medical image analysis methods, specifically designed to deal with the massive amount of extracted features, could be beneficial for the definition of imaging biomarkers, guiding towards personalized patient care. However, these large-scale studies need efficient techniques to drastically reduce the prohibitive running time that is typically required.

In this paper, we presented HaraliCU, a computationally efficient approach capable of effectively exploiting the power of the modern GPUs, which aims at accelerating the GLCM computation by keeping the full dynamic range in medical images. Our method was tested on a dataset composed of brain metastatic tumor MR images and ovarian cancer CT images. Our C++ coded sequential version showed to be  $\sim 200\times$  faster than the corresponding MatLab implementation. In addition, the GPU-powered version was able to achieve speed-ups up to  $15.80\times$  and  $19.50\times$ , with respect to the CPU version, on brain metastasis MR and ovarian cancer CT images, respectively. It is worth noting that neither the C++ version nor HaraliCU implementations have been optimized. Indeed, we expect to further increase their performance by exploiting vectorial instructions and multi-threading, in the case of the sequential version, and by carefully using the high-performance memories of the GPU (i.e., registers, shared memory), for what concerns HaraliCU.

Finally, thanks to this outstanding performance, the C++ version and even more so HaraliCU might enable multi-scale radiomic analyses by properly combining several values of distance offsets, orientations, and window sizes.

As a future development, we plan to develop an improved version of HaraliCU by exploiting the vectorization of the input image matrices for a better GPU thread block managing. In order to improve the scalability of the proposed

approach, the dynamic parallelism, supported by CUDA, could be exploited to further parallelize the computations when the workload increases (e.g., high window size). Moreover, even though the spatial and temporal locality are already exploited during the GLCM construction process, based on the sliding window, the usage of the GPU memory hierarchy might be optimized [39]. Finally, dealing with the clinical feasibility of radiogenomic studies, the integration of the imaging phenotype and genotype can provide valuable information about tumor heterogeneity as well as treatment response [40], by efficiently exploiting high-throughput techniques.

## Acknowledgment

This work was partially supported by The Mark Foundation for Cancer Research and Cancer Research UK Cambridge Centre [C9685/A25177]. Additional support has been provided by the National Institute of Health Research (NIHR) Cambridge Biomedical Research Centre. The views expressed are those of the authors and not necessarily those of the NHS, the NIHR or the Department of Health and Social Care.

## References

1. Trivedi, M.M., Harlow, C.A., Connors, R.W., Goh, S.: Object detection based on gray level cooccurrence. *Comput. Vis. Graph. Image Process.* **28**(2) (1984) 199–219
2. Soh, L.K., Tsatsoulis, C.: Texture analysis of SAR sea ice imagery using gray level co-occurrence matrices. *IEEE Trans. Geosci. Remote Sens.* **37**(2) (1999) 780–795
3. Torheim, T., Malinen, E., Kvaal, K., Lyng, H., Indahl, U.G., Andersen, E.K., Futsaether, C.M.: Classification of dynamic contrast enhanced MR images of cervical cancers using texture analysis and support vector machines. *IEEE Trans. Med. Imaging* **33**(8) (2014) 1648–1656
4. Yankeelov, T.E., Mankoff, D.A., Schwartz, L.H., Lieberman, F.S., Buatti, J.M., Mountz, J.M., Erickson, B.J., Fennessy, F.M., Huang, W., Kalpathy-Cramer, J., et al.: Quantitative imaging in cancer clinical trials. *Clin. Cancer Res.* **22**(2) (2016) 284–290
5. Lambin, P., Rios-Velazquez, E., Leijenaar, R., Carvalho, S., van Stiphout, R.G., Granton, P., Zegers, C.M., Gillies, R., Boellard, R., Dekker, A., et al.: Radiomics: extracting more information from medical images using advanced feature analysis. *Eur. J. Cancer* **48**(4) (2012) 441–446
6. Lambin, P., Leijenaar, R.T., Deist, T.M., Peerlings, J., de Jong, E.E., van Timmeren, J., Sanduleanu, S., Larue, R.T., Even, A.J., Jochems, A., et al.: Radiomics: the bridge between medical imaging and personalized medicine. *Nat. Rev. Clin. Oncol.* **14**(12) (2017) 749
7. Yip, S.S., Aerts, H.J.: Applications and limitations of radiomics. *Phys. Med. Biol.* **61**(13) (2016) R150
8. Stoyanova, R., Takhar, M., Tschudi, Y., Ford, J.C., Solórzano, G., Erho, N., Balagurunathan, Y., Punnen, S., Davicioni, E., Gillies, R.J., et al.: Prostate cancer radiomics and the promise of radiogenomics. *Transl. Cancer Res.* **5**(4) (2016) 432

9. Chen, C.C., DaPonte, J.S., Fox, M.D.: Fractal feature analysis and classification in medical imaging. *IEEE Trans. Med. Imaging* **8**(2) (1989) 133–142
10. Galloway, M.M.: Texture analysis using gray level run lengths. *Comput. Graph. Image Process.* **4**(2) (1975) 172–179
11. Thibault, G., Fertil, B., Navarro, C., Pereira, S., Cau, P., Levy, N., Sequeira, J., Mari, J.L.: Shape and texture indexes application to cell nuclei classification. *Int. J. Pattern Recognit. Artif. Intell.* **27**(01) (2013) 1357002
12. Zhu, H., Goodyear, B., Lauzon, M., Brown, R., Mayer, G., Law, A., Mansinha, L., Mitchell, J.R.: A new local multiscale Fourier analysis for medical imaging. *Med. Phys.* **30**(6) (2003) 1134–1141
13. Arivazhagan, S., Ganesan, L.: Texture classification using wavelet transform. *Pattern Recognit. Lett.* **24**(9-10) (2003) 1513–1521
14. Haralick, R.M., Shanmugam, K., Dinstein, I.: Textural features for image classification. *IEEE Trans. Syst. Man Cybern.* **SMC-3**(6) (1973) 610–621
15. Haralick, R.M.: Statistical and structural approaches to texture. *Proc. IEEE* **67**(5) (1979) 786–804
16. Brynolfsson, P., Nilsson, D., Torheim, T., Asklund, T., Karlsson, C.T., Trygg, J., Nyholm, T., Garpebring, A.: Haralick texture features from apparent diffusion coefficient (ADC) MRI images depend on imaging and pre-processing parameters. *Sci. Rep.* **7**(1) (2017) 4041
17. Gómez, W., Pereira, W., Infantosi, A.F.C.: Analysis of co-occurrence texture statistics as a function of gray-level quantization for classifying breast ultrasound. *IEEE Trans. Med. Imaging* **31**(10) (2012) 1889–1899
18. Ortiz, A., Górriz, J., Ramírez, J., Salas-Gonzalez, D., Llamas-Elvira, J.M.: Two fully-unsupervised methods for MR brain image segmentation using SOM-based strategies. *Appl. Soft Comput.* **13**(5) (2013) 2668–2682
19. Park, S., Kim, B., Lee, J., Goo, J.M., Shin, Y.G.: GGO nodule volume-preserving nonrigid lung registration using GLCM texture analysis. *IEEE Trans. Biomed. Eng.* **58**(10) (2011) 2885–2894
20. Rundo, L., Tangherloni, A., Nobile, M.S., Militello, C., Besozzi, D., Mauri, G., Cazzaniga, P.: MedGA: a novel evolutionary method for image enhancement in medical imaging systems. *Expert Syst. Appl.* **119** (2019) 387–399
21. Derle, L., Ammari, S., Bateson, M., Durand, P.B., Haspinger, E., Massard, C., Jaudet, C., Varga, A., Deutsch, E., Soria, J.C., et al.: Limits of radiomic-based entropy as a surrogate of tumor heterogeneity: ROI-area, acquisition protocol and tissue site exert substantial influence. *Sci. Rep.* **7**(1) (2017) 7952
22. Gipp, M., Marcus, G., Harder, N., Suratane, A., Rohr, K., König, R., Männer, R.: Haralicks texture features computation accelerated by GPUs for biological applications. In: *Modeling, Simulation and Optimization of Complex Processes*. Springer (2012) 127–137
23. Leijenaar, R.T., Nalbantov, G., Carvalho, S., Van Elmpt, W.J., Troost, E.G., Boellaard, R., Aerts, H.J., Gillies, R.J., Lambin, P.: The effect of SUV discretization in quantitative FDG-PET radiomics: the need for standardized methodology in tumor texture analysis. *Sci. Rep.* **5** (2015) 11075
24. Orlhac, F., Soussan, M., Chouahnia, K., Martinod, E., Buvat, I.: 18F-FDG PET-derived textural indices reflect tissue-specific uptake pattern in non-small cell lung cancer. *PLoS One* **10**(12) (2015) e0145063
25. Orlhac, F., Soussan, M., Maisonobe, J.A., Garcia, C.A., Vanderlinden, B., Buvat, I.: Tumor texture analysis in 18F-FDG PET: relationships between texture parameters, histogram indices, standardized uptake values, metabolic volumes, and total lesion glycolysis. *J. Nucl. Med.* **55**(3) (2014) 414–22

26. Jen, C.C., Yu, S.S.: Automatic detection of abnormal mammograms in mammo-graphic images. *Expert Syst. Appl.* **42**(6) (2015) 3048–3055
27. Shafiq-ul Hassan, M., Latifi, K., Zhang, G., Ullah, G., Gillies, R., Moros, E.: Voxel size and gray level normalization of CT radiomic features in lung cancer. *Sci. Rep.* **8**(1) (2018) 10545
28. Larue, R.T., van Timmeren, J.E., de Jong, E.E., Feliciani, G., Leijenaar, R.T., Schreurs, W.M., Sosef, M.N., Raat, F.H., van der Zande, F.H., Das, M., et al.: Influence of gray level discretization on radiomic feature stability for different CT scanners, tube currents and slice thicknesses: a comprehensive phantom study. *Acta Oncol.* **56**(11) (2017) 1544–1553
29. Luebke, D.: CUDA: scalable parallel programming for high-performance scientific computing. In: *Proc. 5th IEEE International Symposium on Biomedical Imaging: From Nano to Macro (ISBI), IEEE* (2008) 836–838
30. Nobile, M.S., Cazzaniga, P., Tangherloni, A., Besozzi, D.: Graphics processing units in bioinformatics, computational biology and systems biology. *Brief. Bioinform.* **18**(5) (2016) 870–885
31. Eklund, A., Dufort, P., Forsberg, D., LaConte, S.M.: Medical image processing on the GPU—past, present and future. *Med. Image Anal.* **17**(8) (2013) 1073–1094
32. Smistad, E., Falch, T.L., Bozorgi, M., Elster, A.C., Lindseth, F.: Medical image segmentation on GPUs—a comprehensive review. *Med. Image Anal.* **20**(1) (2015) 1–18
33. Shen, D., Wu, G., Suk, H.I.: Deep learning in medical image analysis. *Annu. Rev. Biomed. Eng.* **19** (2017) 221–248
34. Tsai, H.Y., Zhang, H., Hung, C.L., Min, G.: GPU-accelerated features extraction from magnetic resonance images. *IEEE Access* **5** (2017) 22634–22646
35. Militello, C., Rundo, L., Vitabile, S., Russo, G., Pisciotta, P., Marletta, F., Ippolito, M., D’Arrigo, C., Midiri, M., Gilardi, M.C.: Gamma Knife treatment planning: MR brain tumor segmentation and volume measurement based on unsupervised fuzzy c-means clustering. *Int. J. Imaging Syst. Technol.* **25**(3) (2015) 213–225
36. Vargas, H.A., Veeraraghavan, H., Micco, M., Nougaret, S., Lakhman, Y., Meier, A.A., Sosa, R., Soslow, R.A., Levine, D.A., Weigelt, B., et al.: A novel representation of inter-site tumour heterogeneity from pre-treatment computed tomography textures classifies ovarian cancers by clinical outcome. *Eur. Radiol.* **27**(9) (2017) 3991–4001
37. Rizzo, S., Botta, F., Raimondi, S., Origgi, D., Buscarino, V., Colarieti, A., Tomao, F., Aletti, G., Zanagnolo, V., Del Grande, M., et al.: Radiomics of high-grade serous ovarian cancer: association between quantitative CT features, residual tumour and disease progression within 12 months. *Eur. Radiol.* (2018) 1–11
38. Pinker, K., Shitano, F., Sala, E., Do, R.K., Young, R.J., Wibmer, A.G., Hricak, H., Sutton, E.J., Morris, E.A.: Background, current role, and potential applications of radiogenomics. *J. Magn. Reson. Imaging* **47**(3) (2018) 604–620
39. Gupta, S., Xiang, P., Zhou, H.: Analyzing locality of memory references in GPU architectures. In: *Proc. ACM SIGPLAN Workshop on Memory Systems Performance and Correctness, ACM* (2013) 12
40. Sala, E., Mema, E., Himoto, Y., Veeraraghavan, H., Brenton, J.D., Snyder, A., Weigelt, B., Vargas, H.A.: Unravelling tumour heterogeneity using next-generation imaging: radiomics, radiogenomics, and habitat imaging. *Clin. Radiol.* **72**(1) (2017) 3–10