2025 IEEE/ACM International Symposium on Low Power Electronics and Design (ISLPED) | 979-8-3315-2710-5/25/$31.00 ©2025 IEEE | DOI: 10.1109/ISLPED65674.2025.11261773

Power Map Characterization and Modeling for Commercial CPU/GPUs Considering Temperature Dependence

Jincong Lu, Sachin Sachdeva, Haotian Lu, and Sheldon X.-D. Tan
Department of Electrical and Computer Engineering, University of California, Riverside, CA 92521 USA
jincong.lu@email.ucr.edu, ssach008@ucr.edu, hlu123@ucr.edu, stan@ece.ucr.edu

***Abstract*—In this paper, we address the challenge of accurate full-chip power mapping for commercial off-the-shelf CPU and GPU processors, explicitly considering temperature dependence. It is well known that both dynamic and leakage power are strongly temperature-dependent; however, existing power estimation methods for real chips often neglect this critical factor. To mitigate this, we characterize temperature-dependent spatial power maps for the first time on commercial processors, including the AMD Radeon RX 6400 GPU and Qualcomm Snapdragon 680 (SM6225) CPU. Using a back-side cooling infrared (IR) thermal imaging system, we capture full-chip thermal maps under different cooling conditions while running identical workloads. These thermal maps are converted into power maps using first-principles-based methods. By repeating this process across varying cooling environments, we collect power maps corresponding to different average chip temperatures. Our experimental results confirm that both total power and spatial power distributions vary significantly with cooling conditions, even under the same workload. We then train machine learning models using real-time performance and utilization metrics—collected via AMD Adrenalin Edition and Qualcomm Snapdragon Profiler—to capture these thermal effects. Two deep neural network architectures are explored: a transformer-based model, *ChipPowerMap*, and a CNN-based decoder model. We compare their performance in accurately predicting temperature-aware full-chip power maps. Numerical results highlight the effectiveness of *ChipPowerMap* in achieving highly accurate thermal map predictions, boasting an RMSE of only $67.88\mathrm{mW/mm^2}$ or 0.97% of the full-scale error. It also outperforms the CNN-based method by $1.62\mathrm{x}$ in terms of accuracy on average. Besides, the proposed model offers real-time estimation with a rapid speed of $25\mathrm{ms}$ on the target chip.**

I. Introduction

With continued integration and scaling, modern high-performance multi/many-core processors are facing increasingly severe thermal challenges. This issue is further amplified by the exponential growth of generative AI technologies like ChatGPT [1], which demand massive computational resources. As a result, thermal and power modeling for commercial AI chips and hardware has become critically important, not only because of their high power consumption, but also due to the substantial cost of cooling solutions. Elevated temperatures significantly reduce chip reliability [2], especially in AI processors such as NVIDIA's GPUs, where power consumption can exceed 1000W. For instance, the H100/H200 GPUs have Thermal Design Powers (TDPs) of 700W, while the latest B200 reaches as high as 1200W [3]. To address this, runtime power and thermal management schemes have become an essential part of processor design [4], [5]. These control strategies rely on accurate, real-time thermal information, ideally in the form of a spatial thermal map [6], [7]. However, due to strict area and power constraints, on-chip sensors are limited in number and cannot provide full-chip coverage [8].

Recent advances have introduced machine learning-based approaches for full-chip thermal mapping and hotspot detection in commercial multi-core processors, leveraging the high modeling capacity of deep neural networks [9]–[12]. These techniques utilize real-time chip performance and utilization metrics—such as frequency, voltage, instructions per cycle (IPC), cache usage, and various performance counters, which are readily accessible through on-chip monitoring infrastructure [13]. Profiling tools like Intel PCM [14] and AMD uProf [15] further facilitate data collection. Early efforts employed LSTM models for low-latency online thermal estimation [9], [10], followed by GAN-based methods for synthesizing spatial thermal profiles [11]. More recently, transformer-based models have been developed and tailored for both AMD CPUs [12] and NVIDIA GPUs [16]. Collectively, these approaches mark a transition toward data-driven thermal inference, offering high spatial accuracy and adaptability during runtime.

However, existing machine learning models in prior work are primarily trained using thermal maps obtained through direct infrared (IR) imaging of processors without cooling sinks—a setup required by IR imaging but one that fails to reflect real operating conditions. This mismatch limits the practical applicability of those models. To address this challenge, a more realistic approach involves first estimating power density maps under actual workloads, and then computing thermal maps using accurate thermal models and realistic boundary conditions that account for practical cooling configurations. Some early efforts have explored this power-density-based thermal estimation strategy [17]. However, these methods overlook the temperature dependence of power consumption and do not establish a direct machine learning pipeline from real-time performance metrics to power maps.

In this work, we aim to address the aforementioned limitation by characterizing the temperature-dependent spatial power maps of commercial processors—specifically, an AMD GPU and a Qualcomm CPU—for the first time. To achieve this, we first capture real-time, full-chip thermal maps using a back-side cooling IR imaging system, which enables accurate thermal measurement under active cooling. We then convert these thermal maps into corresponding power maps using first-

The work is supported in part by NSF grant under No.CCF-2007135, and in part by NSF grant under No. CCF-2113928.

principles-based thermal modeling methods. By repeating this process under different cooling conditions while keeping the workloads constant, we collect thermal and power maps across a range of average temperatures. Our data clearly show that even under identical workloads, both the total power consumption and spatial power distribution of real commercial processors vary significantly with cooling conditions—highlighting the strong temperature dependence of power behavior in practical scenarios. Fig. 1 shows the average temperature and power of AMD Radeon RX 6400 GPU under the same workload and different cooling conditions. We can see that as the temperature increases by more than ten degrees, the GPU power increases from 8.99 watts to 11.71 watts, even though the workload is exactly the same.

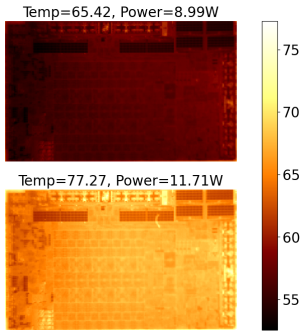


Fig. 1: Thermal map of AMD Radeon RX 6400 under the same workload and different cooling conditions

The key contributions of this study are as follows.

- First, we managed to obtain real-time full-chip thermal maps for two commercial multi/many core processors—an AMD Radeon RX 6400 GPU and a Qualcomm Snapdragon 680 (SM6225) CPU—using a back-side cooling IR thermal imaging system without a heat sink. We then applied first-principles-based methods to convert these thermal maps into corresponding power maps. In total, we collected $19,655$ pairs of performance metrics and thermal maps for the AMD GPU, and $79,707$ pairs for the Qualcomm CPU.
- We then repeated the thermal imaging process under varying cooling conditions while keeping the workloads constant, allowing us to capture chip behavior at different average temperatures. For the first time, we observed that in commercial processors such as the AMD Radeon RX 6400 GPU, both the total power consumption and the spatial power distribution vary significantly under identical workloads when cooling conditions change. This key observation motivates the development of new machine learning models that explicitly account for temperature-dependent power behavior.
- To incorporate temperature-dependent effects into machine learning-based power modeling, we construct spatial power maps under a range of cooling conditions and use them for training. The models are trained on real-time performance and resource utilization metrics collected from two commercial multi-/many-core processors: the AMD Radeon RX 6400 GPU and the Qualcomm Snapdragon 680 (SM6225) CPU. Metric collection is performed using AMD Software: Adrenalin Edition for the RX6400 and Snapdragon Profiler for the SM6225. We evaluate two deep neural network architectures for power map estimation: a transformer-based model, referred to as *ChipPowerMap*, and a CNN-based decoder model. Their performance is compared to assess effectiveness in capturing temperature-dependent spatial power characteristics.
- Numerical results demonstrate the high accuracy of the power map predictions, with a root-mean-square error of only $67.88\mathrm{mW}/\mathrm{mm}^2$ or 0.97% of the full-scale error. Also, our *ChipPowerMap* outperforms the CNN-based method by $1.62\mathrm{x}$ in terms of accuracy on average, proving that the transformer-based method is superior to the CNN-based method in the scenario of power map prediction through time series input. Furthermore, the proposed approach can be deployed on the target chip with a fast inference speed of $25\mathrm{ms}$, making it suitable for real-time estimation.

This article is organized as follows. Section II provides a review of relevant work. Section III outlines the thermal modeling framework and IR thermography setup employed in this study. Section IV explains the process of collecting and preparing the training thermal data, as well as the selection of performance metrics features for the proposed method. Section V describes the architecture of the proposed transformer-based model for thermal map estimation. Section VI presents the experimental results and provides comparisons. Section VII concludes the article.

II. Related work

Power modeling in post-silicon stage aims to estimate power consumption at the granularity of functional blocks or to reconstruct full-chip power density maps under various workloads. Early efforts have attempted to derive component-wise and total power for real microprocessors [18]–[21]. A common approach involves adjusting per-unit power values until their sum aligns with experimentally measured total power [18], [19], though this often relies on manual tuning. Wu et al. [20] addressed this challenge by applying linear regression and K-means clustering to identify characteristic power patterns. Dev et al. [21] formulated the estimation as a constrained optimization problem, incorporating thermal models obtained via finite-element simulation and empirical data.

In parallel, many studies have explored post-silicon full-chip power map estimation [22]–[27]. These typically pose the inverse thermal-to-power estimation as a nonlinear optimization problem:

$$\min ||M \cdot \mathbf{p} - \mathbf{t}||^2 \tag{1}$$

where M is the power-to-temperature transfer matrix derived from thermal modeling, $\mathbf{p}$ is the power density vector, and $\mathbf{t}$ is the measured or simulated temperature vector at discrete die locations. The matrix M may be obtained from FPGA-based calibration [23]–[25], [27], from power blurring techniques [22], or through parameterized analytic models combined with regression [26]. Reda et al. [27] demonstrated this approach on commercial multi-core processors, though their method provides

only core-level granularity based on aggregated power and sensor data.

Paek et al. [24] introduced a statistical perspective by formulating power estimation as a maximum likelihood problem, estimating power distribution $\mathbf{p}$ conditioned on a given thermal map $\mathbf{t}$. However, this approach relies heavily on accurate thermal modeling, typically using simulation tools like HotSpot [28] to generate reliable baseline data. Although the method was evaluated on an FPGA platform, it achieved only 90.7% accuracy on average, highlighting the inherent difficulty in constructing precise thermal models for real silicon.

These existing methods are generally not suitable for off-the-shelf commercial multi-core processors, as most have been developed and validated on specialized silicon platforms such as FPGAs [23]–[25], [27] or custom 3D ICs [26]. Moreover, they often incur significant computational overhead due to the need to solve complex nonlinear inverse problems, as formulated in Eq.(1).

On the other hand, recent studys show that the relative power density map can be easily obtained by 2D spatial Laplace transformation of measured or calculated temperature maps based on the first principle of heat conduction [8], [29]. This work has been extended to calculate the real power map of commercial CPUs [17]. This work has been extended to model power density for TPU [30] recently. But this method does not consider the temperature dependence of chip power consumption, which is important as obtained power density maps will not be accurate across wide temperature ranges under various workloads. It also failed to build the machine learning models directly from the real-time performance metrics to resulting power maps.

Conversely, machine learning-based approaches offer new perspectives on real-time full-chip thermal map estimation methods for commercial multi-core CPUs. These methods utilize real-time on-chip utilization and monitoring information as inputs for generating thermal and power maps. Such ideas have been explored recently by DNN-based approaches leveraging real-time performance metrics [9]–[12]. Sadiqbatcha et al. first proposed an LSTM-based approach called Realmaps, which utilizes Intel PCM metrics for estimating full-chip thermal maps in commercial off-the-shelf multi-core processors [9], [10]. This approach has shown promising results in terms of accuracy and speed of inference for real-time applications. On top of this, an improvement was made by employing an image-friendly CNN model based on the GAN architecture. This approach, known as ThermGAN, demonstrates better results than the LSTM-based methods in terms of accuracy [11]. Recently, a transformer-based DNN method has been proposed to estimate the full-chip thermal map of AMD multicore chips using uProf utilization metrics [12]. This method exhibits superiority over the GAN-based and LSTM-based methods due to its powerful modeling capability for time series data via the attention mechanism. This work has been extended to model the thermal maps for NVIDIA GPUs [16]. However, most of the machine learning models developed thus far were trained using thermal maps obtained from processors without cooling solutions, which do not accurately reflect real operating conditions due to the constraints of thermal IR imaging.

III. Power Density Map Estimation Framework

A. Estimation flow overview

The proposed approach consists of three main components. First, we collect data by logging real-time processor metrics during workload execution. At the same time, we use an IR imaging system to capture full-chip thermal maps, from which we compute corresponding power density maps. These datasets form the foundation for training a transformer-based model for online power prediction. Fig. 2 illustrates the overall framework of the proposed approach.

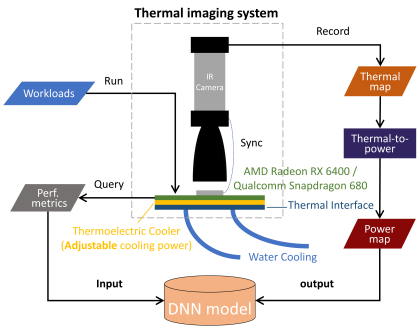


Fig. 2: Framework and data acquisition flow

B. Thermal IR imaging system

To measure the chip surface temperature map, we deploy an advanced IR thermal imaging system, as illustrated in Fig. 3. However, the front surface of the processor's core module is typically covered by a heat sink, making it inaccessible for thermal imaging. To overcome this, we adopt a back-side liquid cooling system [31] in place of conventional front-side cooling. Since back-side cooling requires heat to pass through the PCB, its thermal efficiency is inherently lower. To compensate for this, a thermoelectric cooler (TEC) device is installed beneath the processor module on the PCB, enhancing heat extraction from the back side. a TEC is a solid-state active heat pump that transfers heat from one side of the device to the other when an electric current passes through it. Its heat transfer power can be controlled by adjusting the current.

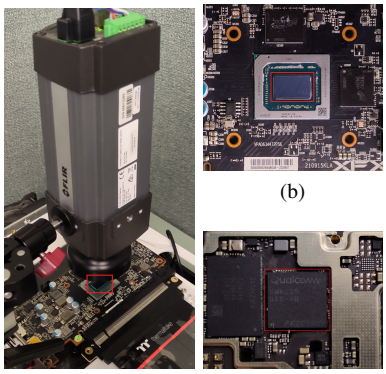
The model of the IR camera is FLIR A325sc. It can capture thermal images with a maximum resolution of 240×320 pixels (px) at a maximum frequency of 60Hz. The factory-calibrated IR sensor ensures accuracy within a temperature range of $-20°\mathrm{C}$ to $120°\mathrm{C}$ and resolves the IR spectral range of $7.5\mu m$ to $13\mu m$.

IV. Data Preparation and Performance Metrics Selection

A. Thermal map acquisition

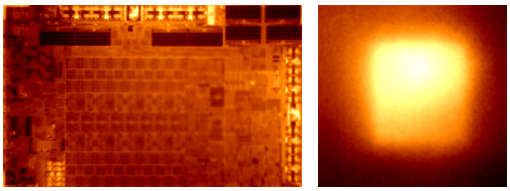
In this work, an AMD Radeon RX 6400 GPU chip and a Qualcomm Snapdragon 680 (SM6225) chip are studied. Fig. 4 shows the thermal map examples.

To study the impact of different cooling conditions and temperatures, we adjust the active cooling strength by varying the current supplied to the TEC device. Five different current levels are tested while running the same tasks. The final



(a) (b) (c)

Fig. 3: (a) Thermal imaging system setup (b) GPU chip under-test, AMD RX6400 (c) CPU chip under-test, Qualcomm SM6225. Core module is shown in the red box.



(a) (b)

Fig. 4: The thermal image examples of (a) AMD Radeon RX 6400 (b) Qualcomm Snapdragon 680 (SM6225)

collected dataset thus covers a range of scenarios under varying cooling conditions.

B. Performance metrics acquisition

AMD provides a management and monitoring tool for their GPUs, known as AMD Software: Adrenalin Edition. Through this utility, we can query the device state and gather GPU metrics to obtain information about the performance and thermal behavior of the GPU. For the Qualcomm chip, Snapdragon Profiler is utilized.

Table I provides a list of the metrics we selected, including the readings from sensors such as the temperature, as well as metrics on current memory usage, frequency, and more. In total, we have 10 metrics for the RX6400 and 105 metrics for the SM6225.

Performance metrics have a minimum collection interval, which is much lower than the sampling rate of the thermal camera. The final data collection frequency is determined by the sampling frequency of the performance metrics. AMD Software has a maximum sampling frequency of 4 Hz, while Snapdragon Profiler can reach up to 18 Hz.

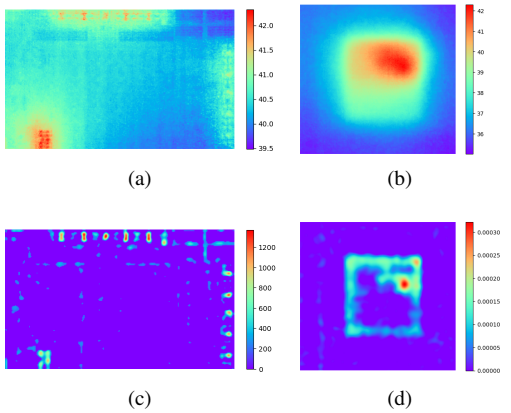
We use the historical time series of metric vectors from the previous 10 frames as input for thermal prediction, making the processing part of this similar to that of natural language tasks.

C. Power map acquisition from measured thermal maps

There have been various post-silicon approaches transforming thermal distribution to power distribution [22]–[27], [32].

TABLE I: Selected Performance Metrics

AMD Radeon RX 6400		
GPU Util	GPU SCLK	GPU Power
GPU Temp	GPU Hotspot Temp	GPU Voltage
GPU Mem Util	GPU MCLK	CPU Util
System Mem Util		
Qualcomm Snapdragon 680 (SM6225)		
CPU Frequency 0-7	CPU Load 0-7	CPU % Utilization 0-7
% Global Buffer Read L2 Hit	% Global Buffer Write L2 Hit	% Image Read L2 Hit
% Kernel Load Cycles	% L1 Hit	Avg Load-Store Instructions Per Cycle
Bytes Data Actually Written	Bytes Data Write Requested	Global Buffer Data Read BW
Global Buffer Data Read Request BW	Global Image Compressed Data Read BW	Global Image Data Read Request BW
Global Image Uncompressed Data Read BW	Global Memory Atomic Instructions	Global Memory Load Instructions
Global Memory Store Instructions	Load-Store Utilization	Local Memory Atomic Instructions
Local Memory Load Instructions	Local Memory Store Instructions	Clocks / Second
GPU % Bus Busy	Avg Bytes / Fragment	Avg Bytes / Vertex
Read Total	SP Memory Read	Texture Memory Read BW
Vertex Memory Read	Write Total	Avg Preemption Delay
Preemptions / second	% Prims Clipped	% Prims Trivially Rejected
Average Polygon Area	Average Vertices / Polygon	Pre-clipped Polygons/Second
Reused Vertices / Second	% Anisotropic Filtered	% Linear Filtered
% Nearest Filtered	% Non-Base Level Textures	% Shader ALU Capacity Utilized
% Shaders Busy	% Shaders Stalled	% Texture Pipes Busy
% Time ALUs Working	% Time Compute	% Time EFUs Working
% Time Shading Fragments	% Time Shading Vertices	ALU / Fragment
ALU / Vertex	EFU / Fragment	EFU / Vertex
Fragment ALU Instructions / Sec (Full)	Fragment ALU Instructions / Sec (Half)	Fragment EFU Instructions / Second
Fragment Instructions / Second	Fragments Shaded / Second	Textures / Fragment
Textures / Vertex	Vertex Instructions / Second	Vertices Shaded / Second
% Instruction Cache Miss	% Stalled on System Memory	% Texture Fetch Stall
% Texture L1 Miss	% Texture L2 Miss	% Vertex Fetch Stall
L1 Texture Cache Miss Per Pixel	Download Mbps (WiFi)	Rx Bytes (WiFi)
Rx Packets (WiFi)	Tx Bytes (WiFi)	Tx Packets (WiFi)
Upload Mbps (WiFi)	Available Memory	Free Memory
Total Memory	Used Memory	Temperature



(a) (b) (c) (d)

Fig. 5: (a) RX6400 thermal image (°C). (b) SM6225 thermal image (°C). (c) RX6400 power map ($\mathrm{mW/mm^2}$). (d) SM6225 power map ($\mathrm{mm^{-2}}$).

Among those [8], [32] suits for our study case best, giving it calculates spatially continuous and relatively precise power maps from thermal maps with high efficiency, which is suitable for real-time inferences.

Considering the steady state 2D spatial thermal distribution of the processor as $T(x, y)$, where (x, y) is the coordinates of the thermal map. The power density map can be approximated as [32]:

$$p(x,y) \approx \begin{cases} k[-\nabla^2 T(x,y)], & -\nabla^2 T(x,y) > 0 \\ 0, & -\nabla^2 T(x,y) \leq 0 \end{cases} \tag{2}$$

with

$$k = \kappa \Delta z \tag{3}$$

where $p(x, y)$ stands for the spatial power map (density,

Watt/area), κ and Δz for thermal conductivity and chip thickness, which are constants. And $\nabla^2 T(x,y)$ is the 2D Laplacian of temperature. The coefficient k is expressed by:

$$k = \kappa \Delta z \approx \frac{P}{-\int_{S_P} \nabla^2 T(x,y) dxdy} \tag{4}$$

where S_P indicates the area where the negative-Laplacian term of temperature $[-\nabla^2 T(x,y)]$ is positive. The negative-Laplacian term reflects the pattern of spatial power distribution. In this work, we call k the thermal-to-power coefficient. It can be calculated by the thermal measurement $T(x,y)$ combined with total power consumption P provided as one of the performance metrics. Once we have $T(x,y)$ and P, we can substitute them for Eq. (4) to obtain k. After k is obtained, power density maps can be acquired straightforwardly through Eq. (2).

Sometimes, the processor does not provide total power consumption as one of its performance metrics, for example, SM6225 in this study. In such cases, we cannot obtain an estimate of the absolute power density, but we can still derive an estimation of the relative power distribution:

$$p_r(x,y) = \frac{p(x,y)}{P} \approx \begin{cases} \frac{\nabla^2 T(x,y)}{\int_{S_P} \nabla^2 T(x,y) dxdy}, & -\nabla^2 T(x,y) > 0 \\ 0, & -\nabla^2 T(x,y) \leq 0 \end{cases} \tag{5}$$

where $p_r(x,y)$ stands for the spatial relative power map (area^{-1}). This relative distribution can still help identify hotspots and guide task scheduling and cooling resource allocation.

Although the thermal noise is small relative to the temperature, its Laplacian can be locally larger than the Laplacian of temperature, overshadowing useful information. Previous work applied Discrete Cosine Transform (DCT) [33] to convert images into the frequency domain and then removed high-frequency components for denoising. However, this significantly reduces the effective resolution of the images. As an alternative, we use Gaussian filtering for denoising, which allows us to preserve higher resolution while still extracting meaningful information.

The size of RX6400 and SM6225 is about 8.1×13.3 mm and 10.5×11.1 mm. As described in Section IV-A, the final map size is 168×276 px and 227×237 px. Fig. 5 shows the temperature maps and the corresponding generated power maps for two chips. We can observe that for Sanpdragon 680, heat is primarily generated within a square region at the center of the image. This indicates that the actual chip is located only in that area, and a much larger package is being used. Therefore, to enable more efficient prediction, we further crop the power map to focus on the central region containing the actual chip, resulting in a final size of 125×121 px.

V. ChipPowerMap for Full-Chip Power Estimation

Overall, our task involves generating images from time series data. Prior research in CPU heat map prediction has shown that transformer-based methods outperform alternatives such as LSTM-based and GAN-based models when it comes to processing time series [12], [16], [34]. The transformer architecture leverages self-attention mechanisms, allowing it to access and model all historical data simultaneously in a highly parallelized fashion. This design effectively addresses the challenge of capturing long-range dependencies, making it particularly well-suited for our application.

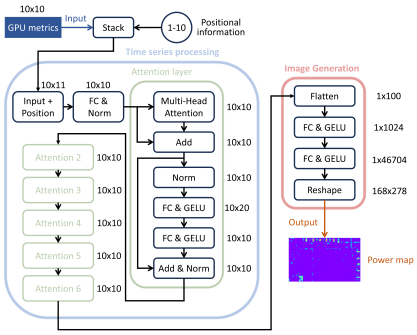


Fig. 6: Architecture of proposed *ChipPowerMap*

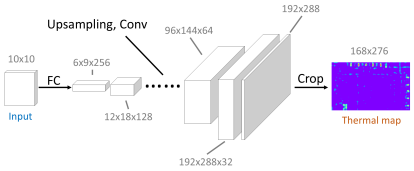


Fig. 7: Architecture of CNN-based model

Fig. 6 illustrates the framework of our model, named *ChipPowerMap*. Since our objective is to generate image outputs, we have not adopted the traditional encoder-decoder architecture but only employ an encoder structure. Subsequently, we use the multi-layer perceptron (MLP) to generate the output and reshape it into a 168×276 px image. This is somewhat akin to GPT-1 [35] only using a decoder structure to predict the next word.

As a comparison, we consider a CNN-based model, as illustrated in Fig. 7. The input time series is first flattened and then passed through a fully connected layer to produce a smaller-sized image with more channels. This intermediate representation is then progressively transformed into the predicted power map through a series of upsampling and convolutional layers.

VI. Experimental Results and Discussions

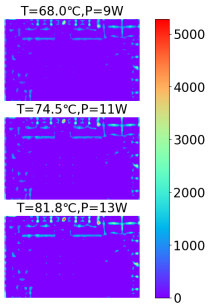
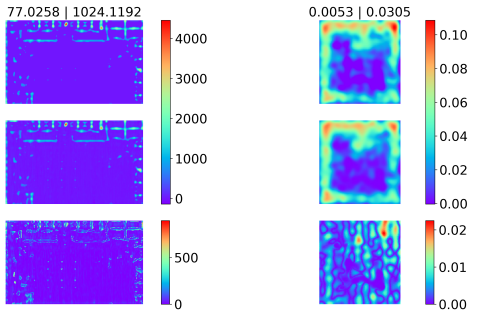


Fig. 8: Power density map of AMD Radeon RX 6400 under the same workload and different cooling conditions

TABLE II: Accuracy

Dataset	Accuracy Term	*ChipPowerMap*	*CNN*	*CNN*/*ChipPowerMap* ratio
RX6400 (AMD)	Average RMSE	$67.88\ \mathrm{mW/mm^2}$	$109.95\ \mathrm{mW/mm^2}$	1.6197
	Maximum RMSE	$212.01\ \mathrm{mW/mm^2}$	$246.48\ \mathrm{mW/mm^2}$	1.1626
	RMSE deviation	$18.06\ \mathrm{mW/mm^2}$	$29.15\ \mathrm{mW/mm^2}$	1.6137
SM6225 (Qualcomm)	Average RMSE	$0.007898\ \mathrm{mm^{-2}}$	$0.009816\ \mathrm{mm^{-2}}$	1.2429
	Maximum RMSE	$0.029066\ \mathrm{mm^{-2}}$	$0.029157\ \mathrm{mm^{-2}}$	1.0031
	RMSE deviation	$0.003951\ \mathrm{mm^{-2}}$	$0.003419\ \mathrm{mm^{-2}}$	0.8652



AMD Radeon RX 6400 Snapdragon 680 SoC

Fig. 9: Measured power maps (row #1), estimated power maps (row #2), and error maps (row #3). The numbers above the maps indicate the *Power density RMSE* | *Average Power*.

We implement *ChipPowerMap* with Python 3.8 and TensorFlow (version 2.11.0) [36], a widely adopted open-source machine learning library. The model was trained on a Linux server equipped with a Xeon E5-2699v4 2.20GHz processor and an NVIDIA Titan RTX GPU. The dataset ultimately comprises $19,655$ data points for RX6400 and $79,707$ data points for SM6225. 80% of the data is allocated for training and 20% is used for testing.

A. Power density map temperature dependency study

We first investigate the impact of temperature on power density maps for the commercial processors. To show such dependency, we manage to keep the processor running under the same workload, while changing the current supplied to the TEC to adjust the active cooling power, which significantly altered the chip's average temperature. Fig. 8 shows the power density maps of the AMD Radeon RX 6400 GPU under three different cooling conditions. As the average temperature increases from 68°C to 82°C, we observe a substantial change in the processor's total power consumption and a notable increase in spatial power density. For every increase of approximately 3°C to 4°C in average temperature, the power consumption increases by about one watt.

B. Power map estimation accuracy study

Next, we investigate the accuracy and efficiency of power map estimation methods proposed for the two commercial processors. Note that for Snapdragon 680 SoC CPU, we use relative power density for the training. The trained models will reflect the temperature dependency if the data has such information. We used the Root-Mean-Square Error (RMSE) between the predicted power map and the ground truth across all pixels except those covered by text to measure accuracy. We compare the transformer-based method *ChipPowerMap* and CNN-based approach.

For RX6400, on the test set, the average RMSE is $67.88\mathrm{mW/mm^2}$, with a standard deviation of only $18.06\mathrm{mW/mm^2}$, across a power density range of 0 to $6992.25\mathrm{mW/mm^2}$. For SM6225, on the test set, the average RMSE is $0.007898\mathrm{mm^{-2}}$, with a standard deviation of $0.003951\mathrm{mm^{-2}}$, across a relative power density range of 0 to $0.248857\mathrm{mm^{-2}}$.

Fig. 9 illustrates the estimated and measured thermal maps, showcasing examples from the test set. Clearly, *ChipPowerMap* has achieved very high precision and the learning of the contour is also very successful.

Furthermore, we compare *ChipPowerMap* with the CNN-based full-chip power map estimation methods in terms of prediction accuracy on the same dataset. The results are shown in Table II. We can see that *ChipPowerMap* is about 1.62x more accurate than CNN-based method on average. The experimental results confirm that in scenarios where the input is a time series, the transformer-based approach indeed exhibits greater advantages in terms of accuracy compared to the CNN-based method.

C. Computational efficiency study

Once the model training is completed, it can be deployed on the target system and the average inference time measured is $25\mathrm{ms}$. This speed ensures that the model can keep up with real-time power map predictions.

VII. Conclusion

In this article, we investigate the impact of temperature dependence on the spatial power distribution of commercial microprocessors. Specifically, we examine two commercial platforms: the AMD Radeon RX 6400 GPU and the Qualcomm Snapdragon 680 (SM6225) CPU. Using a back-side cooling IR thermal imaging system, we capture full-chip thermal maps of these processors under realistic workloads and varying cooling conditions. Power maps are then inferred from the measured thermal data. Our results demonstrate that both total power consumption and spatial power profiles vary significantly with temperature, even under identical workloads. To model this effect, we train machine learning models using real-time performance and utilization metrics extracted from each platform. We evaluate two DNN architectures: a transformer-based model and a CNN-based decoder model. Experimental results show that ChipPowerMap achieves highly accurate power predictions, with a RMSE of only $67.88\mathrm{mW/mm^2}$ (0.97% of full-scale). It also outperforms the CNN-based model by 1.62x in accuracy, while delivering real-time estimation at $25\mathrm{ms}$ inference latency on the target hardware.

REFERENCES

[1] "Chatgpt from openai." [Online]. Available: https://chat.openai.com

[2] "Critical Reliability Challenges for The International Technology Roadmap for Semiconductors (ITRS)," 2003, in International Sematech Technology Transfer Document 03024377A-TR, 2003.

[3] "Nvidia blackwell b200 gpu thermal design power," https://www.nvidia.com/en-us/data-center/dgx-b200/.

[4] H. Esmaeilzadeh, E. Blem, R. St. Amant, K. Sankaralingam, and D. Burger, "Dark silicon and the end of multicore scaling," *Micro, IEEE*, vol. 32, no. 3, pp. 122–134, May 2012.

[5] M. Taylor, "A landscape of the new dark silicon design regime," *IEEE/ACM International Symposium on Microarchitecture*, vol. 33, no. 5, pp. 8–19, October 2013.

[6] K.Skadron, M. R. Stan, W. Huang, S. Velusamy, K. Sankaranarayanan, and D. Tarjan, "Temperature-aware microarchitecture," in *International Symposium on Computer Architecture*, 2003, pp. 2–13.

[7] J. Kong, S. W. Chung, and K. Skadron, "Recent thermal management techniques for microprocessors," *ACM Comput. Surv.*, vol. 44, no. 3, pp. 13:1–13:42, jun 2012. [Online]. Available: http://doi.acm.org/10.1145/2187671.2187675

[8] S. Sadiqbatcha, J. Zhang, H. Zhao, H. Amrouch, J. Henkel, and S. X.-D. Tan, "Post-silicon heat-source identification and machine-learning-based thermal modeling using infrared thermal imaging," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, 2020.

[9] S. Sadiqbatcha, Y. Zhao, J. Zhang, H. Amrouch, J. Henkel, and S. X. D. Tan, "Machine learning based online full-chip heatmap estimation," in *2020 25th Asia and South Pacific Design Automation Conference (ASP-DAC)*, 2020, pp. 229–234.

[10] S. Sadiqbatcha, J. Zhang, H. Amrouch, and S. X.-D. Tan, "Real-time full-chip thermal tracking: A post-silicon, machine learning perspective," *IEEE Transactions on Computers*, 2021.

[11] W. Jin, S. Sadiqbatcha, J. Zhang, and S. X.-D. Tan, "Full-chip thermal map estimation for commercial multi-core cpus with generative adversarial learning," in *Proc. Int. Conf. on Computer Aided Design (ICCAD)*. New York, NY, USA: ACM, Nov. 2020, pp. 1–9.

[12] J. Lu, J. Zhang, and S. X.-D. Tan, "Real-time thermal map estimation for amd multi-core cpus using transformer," in *IEEE/ACM International Conference on Computer Aided Design (ICCAD)*, 2023, pp. 1–7.

[13] K. Zhang, A. Guliani, S. Ogrenci-Memik, G. Memik, K. Yoshii, R. Sankaran, and P. Beckman, "Machine learning-based temperature prediction for runtime thermal management across system components," *IEEE Transactions on Parallel and Distributed Systems*, vol. 29, no. 2, pp. 405–419, Feb 2018.

[14] Intel, "Intel Performance Counter Monitor (PCM)," https://software.intel.com/en-us/articles/intel-performance-counter-monitor.

[15] AMD, "AMD uProf software profiling tool," https://developer.amd.com/amd-uprof/.

[16] J. Lu, S. Sachdeva, Y. Lin, and S. X.-D. Tan, "Real-time thermal map characterization and analysis for commercial gpus with ai workloads," in *Proc. Int. Symposium. on Quality Electronic Design (ISQED)*, 2025.

[17] J. Zhang, S. Sadiqbatcha, M. O'Dea, H. Amrouch, and S. X.-D. Tan, "Full-chip power density and thermal map characterization for commercial microprocessors under heat sink cooling," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, pp. 1–1, 2021.

[18] R. Joseph and M. Martonosi, "Run-time power estimation in high-performance microprocessors," in *Proc. Int. Symp. on Low Power Electronics and Design (ISLPED)*, 2001, pp. 135–140.

[19] C. Isci and M. Martonosi, "Runtime power monitoring in high-end processors: Methodology and empirical data," in *Proceedings of MICRO*, 2003.

[20] W. Wu, L. Jin, J. Yang, P. Liu, and S. X.-D. Tan, "Efficient power modeling and software thermal sensing for runtime temperature monitoring," *ACM Trans. on Design Automation of Electronics Systems*, vol. 12, no. 3, pp. 1–29, 2007.

[21] K. Dev, A. N. Nowroz, and S. Reda, "Power mapping and modeling of multi-core processors," in *International Symposium on Low Power Electronics and Design (ISLPED)*, Sept 2013, pp. 39–44.

[22] X. Wang, S. Farsiu, P. Milanfar, and A. Shakouri, "Power trace: An efficient method for extracting the power dissipation profile in an ic chip from its temperature map," *IEEE Transactions on Components and Packaging Technologies*, vol. 32, no. 2, pp. 309–316, 2009.

[23] R. Cochran, A. N. Nowroz, and S. Reda, "Post-silicon power characterization using thermal infrared emissions," in *Proc. Int. Symp. on Low Power Electronics and Design (ISLPED)*. New York, NY, USA: ACM, 2010, pp. 331–336. [Online]. Available: http://doi.acm.org/10.1145/1840845.1840914

[24] S. Paek, W. Shin, J. Sim, and L. Kim, "Powerfield: A probabilistic approach for temperature-to-power conversion based on markov random field theory," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 32, no. 10, pp. 1509–1519, 2013.

[25] A. Nowroz, G. Woods, and S. Reda, "Power mapping of integrated circuits using ac-based thermography," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 21, no. 08, pp. 1398–1409, aug 2013.

[26] F. Beneventi, A. Bartolini, P. Vivet, and L. Benini, "Thermal analysis and interpolation techniques for a logic+ wideio stacked dram test chip," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 35, no. 4, pp. 623–636, 2016.

[27] S. Reda, K. Dev, and A. Belouchrani, "Blind identification of thermal models and power sources from thermal measurements," *IEEE Sensors Journal*, vol. 18, no. 2, pp. 680–691, 2018.

[28] W. Huang, S. Ghosh, S. Velusamy, K. Sankaranarayanan, K. Skadron, and M. R. Stan, "HotSpot: A compact thermal modeling methodology for early-stage VLSI design," *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, vol. 14, no. 5, pp. 501–513, May 2006.

[29] S. Sadiqbatcha, H. Zhao, H. Amrouch, J. Henkel, and S. X.-D. Tan, "Hot spot identification and system parameterized thermal modeling for multi-core processors through infrared thermal imaging," in *2019 Design, Automation Test in Europe Conference Exhibition (DATE)*, March 2019.

[30] J. Lu, J. Zhang, W. Jin, S. Sachdeva, and S. X.-D. Tan, "Learning based spatial power characterization and full-chip power estimation for commercial tpus," in *Proceedings of the 28th Asia and South Pacific Design Automation Conference*, ser. ASPDAC '23. New York, NY, USA: Association for Computing Machinery, 2023, pp. 98–103. [Online]. Available: https://doi.org/10.1145/3566097.3568347

[31] H. Amrouch and J. Henkel, "Lucid infrared thermography of thermally-constrained processors," in *2015 IEEE/ACM International Symposium on Low Power Electronics and Design (ISLPED)*, July 2015, pp. 347–352.

[32] J. Zhang, S. Sadiqbatcha, W. Jin, and S. X. . Tan, "Accurate power density map estimation for commercial multi-core microprocessors," in *Proc. European Design and Test Conf. (DATE)*, 2020, pp. 1085–1090.

[33] N. Ahmed, T. Natarajan, and K. R. Rao, "Discrete cosine transform," *IEEE Transactions on Computers*, vol. C-23, no. 1, pp. 90–93, Jan 1974.

[34] J. Lu and S. X.-D. Tan, "Thermal map dataset for commercial multi/many core cpu/gpu/tpu," in *Proceedings of the 2024 ACM/IEEE International Symposium on Machine Learning for CAD*, ser. MLCAD '24. New York, NY, USA: Association for Computing Machinery, 2024. [Online]. Available: https://doi.org/10.1145/3670474.3685963

[35] A. Radford and I. Sutskever, "Improving language understanding by generative pre-training," in *arxiv*, 2018.

[36] M. Abadi *et al.*, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015. [Online]. Available: http://tensorflow.org/