Fast Retrieval of Weather Analogues in a Multi-petabytes Archive Using Wavelet-Based Fingerprints

Baudouin Raoult[1](✉), Giuseppe Di Fatta[2], Florian Pappenberger[1], and Bryan Lawrence[2,3,4]

[1] European Centre for Medium-Range Weather Forecasts, Reading, UK
{baudouin.raoult,florian.pappenberger}@ecmwf.int
[2] Department of Computer Science, University of Reading, Reading, UK
G.DiFatta@reading.ac.uk, bryan.lawrence@ncas.ac.uk
[3] Department of Meteorology, University of Reading, Reading, UK
[4] National Centre for Atmospheric Science, Reading, UK

Abstract. Very large climate data repositories provide a consistent view of weather conditions over long time periods. In some applications and studies, given a current weather pattern (e.g. today's weather), it is useful to identify similar ones (weather analogues) in the past. Looking for similar patterns in an archive using a brute force approach requires data to be retrieved from the archive and then compared to the query, using a chosen similarity measure. Such operation would be very long and costly. In this work, a wavelet-based fingerprinting scheme is proposed to index all weather patterns from the archive. The scheme allows to answer queries by computing the fingerprint of the query pattern, then comparing them to the index of all fingerprints more efficiently, in order to then retrieve only the corresponding selected data from the archive. The experimental analysis is carried out on the ECMWF's ERA-Interim reanalyses data representing the global state of the atmosphere over several decades. Results shows that 32 bits fingerprints are sufficient to represent meteorological fields over a 1700 km × 1700 km region and allow the quasi instantaneous retrieval of weather analogues.

Keywords: Climate data repositories
Weather analogues · Information retrieval

1 Introduction

Weather analogues is the term used by meteorologists to referrer to similar weather situations. Usually an analogue for a given location or area and forecast lead time is defined as a past prediction, from the same model, that has similar values for selected features of the current model forecast. Before computer simulations were available, weather analogues were the main tool available to forecasters, which is still a usage today [1]. Analogues can be useful on smaller

© Springer International Publishing AG, part of Springer Nature 2018
Y. Shi et al. (Eds.): ICCS 2018, LNCS 10861, pp. 697–710, 2018.
https://doi.org/10.1007/978-3-319-93701-4_55

scale (≈900 km in radius, [2]) as it is otherwise impossible to identify similar patterns in the past given a limited temporal record e.g. at hemispheric scale, similar states the atmosphere would only be observed every 10^{30} years [3]. Usually the maximum record length available is restricted to under 100 years. Weather analogues have many usages. They are used for downscaling model outputs [4], to assess risks of severe weather [5] or managing weather impacts on railway networks [6].

Analogues require comparison of fields and looking for similar patterns in an archive using a brute force approach requires data to be retrieved from the archive and the compared to the query, using a chosen similarity measure. Such operation would be very long and costly on large archive systems as data will typically have to be recalled a tape system.

The aim of this research is to consider an algorithm to index all weather patterns from the archive using a fingerprinting scheme. Queries would be done by computing the fingerprint of the query pattern, then comparing them to the index of all fingerprints, in order to then retrieve the corresponding data from the archive. The main user requirements of such system are:

- the system should be queryable: given a user provided query, the system should return the most similar weather situation from the archive;
- the system should be fast: replies should be perceived by users as “instantaneous”, allowing interactive use;
- newly archived data should be added to the index, without the need to retune/retrain the system.

Wavelet fingerprinting has been successfully used to retrieve images [7] and sounds [8]. The objectives of this paper are therefore to introduce an efficient wavelet fingerprinting system for the retrieval of weather analogues. Efficiency here means that the computation of fingerprint is fast, that the resulting fingerprint is small, that fingerprints can be compared quickly and that they can be stored in an efficient data structure. The fingerprinting method has to be accurate as possible, i.e. that returns the “closest” matching weather according to some agreed similarity measure.

2 Related Work

As the world is generating more and more data, efficient information retrieval has become a major challenge, and is therefore a very active field of research. Information is not only limited to text, but also comprises images, movies and sound. There are many methods available to implement such systems [9,10].

The retrieval system proposed in this work is based on wavelets [11,12], which are expected to capture well the wave-like nature of the weather phenomenon. Wavelets are traditionally use for imagery [13–15], in particular compression [16–20] and image retrieval [7,21,22]. Wavelets have also been used to retrieve medical images [23,24], proteins [25], power management [26–28], time-series analysis [29,30] and image similarity [22,23].

This work builds on the results presented by [7,8], which use wavelets-based algorithms for multi-resolution image querying and audio fingerprinting respectively.

3 The ECMWF Data Archive

The European Centre for Medium-Range Weather Forecasts (ECMWF) has been collecting meteorological information since 1980 and its archive has recently reach over 260 petabytes of primary data. ECMWF's archive is referred to as the Meteorological Archiving and Retrieval System (MARS) [31,32]. This archive provides datasets that covers several decades at hourly temporal resolutions. Because of the size of the archive, most of the data is held on tape, therefore only solutions that do not require access to the data are considered.

The MARS archive contains fields, that are the typical output of numerical weather prediction systems. These are usually gridded data, either global or regional. The grids are sets of regularly distributed points (e.g. one grid point every 5 km) over a given area. Model outputs are collections of fields, one for each variable represented, for a given time and horizontal layer: at large scales (greater than 10 km), the interactions between the different layers of the atmosphere are small compared to the effects of large structures and can be ignored. This is why traditionally meteorologists tend to consider fields are being 2D, their vertical coordinate being an attribute of the field, as is time. Fields are therefore a collection of floating point values geographically distributed according to a mesh (called grid). Most of the grids are regularly spaced.

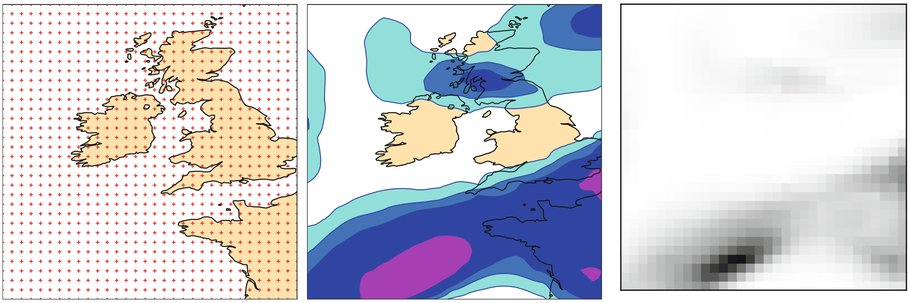
This research will make use of a particular subset of fields so called reanalysis data: a reanalysis is a process by which the same data assimilation system is run on past observations (e.g. over one hundred years), and produces a consistent dataset representing the state of the atmosphere over long periods. This is used for studies linked to climate change [33,34]. These datasets are very well structured and can be easily processed. The data used in this work are selected from the ERA-Interim dataset [35,36], a reanalysis covering the period 1979 to 2014, at 0 UTC (13,149 fields per variable).

Meteorological fields are multidimensional fields, with grid points regularly distributed on the surfaces following the shape Earth: at the surface or at set levels (usually isobaric surface). The fields also vary in time. Although these fields are 4D, they are archived as 2D slices (latitude/longitude), so that users can access long time series of a given surface, or a stack of levels. Fields represent one variable (temperature, pressure, precipitations, etc.), with the value of the variable provided at each grid points.

In the case of regular grids, in which grid points can be organised in a 2D matrix (Fig. 1a), one can see the that this fields can easily be considered as a greyscale image (Fig. 1c, assuming values are normalised to the interval 0–255), although they are traditionally plotted using contours (Fig. 1b).

Four surface variables are selected: 2 m temperature, mean sea level surface pressure (or MSL pressure), 10 m wind speed and total precipitations accumulated over 6 h.

The initial work presented here is limited to a square grid 0.5°×0.5° (≈55 km × 55 km) on the domain 60°N 14°W 44.5°N 1.5°E that covers the British Isles (≈1700 km×1700 km, see Fig. 1), which agrees with the radius of 900 Km presented in [2]. The size of the domain will capture synoptic scales weather patterns.


(a) Geographical distribution of the grid points.

(b) Field of total precipitations over 6 hours.

(c) Same field plotted over a gray map.

Fig. 1. Nature of the meteorological field used in this research. In the middle panel, the total precipitation field is plotted using the traditional methods: contouring and shading (isoline are spaced logarithmically from 0.4 mm to 100 mm.

4 Definition of a Fingerprinting Scheme

4.1 Fingerprinting

The method proposed is to define the fingerprint F of a meteorological field f as:

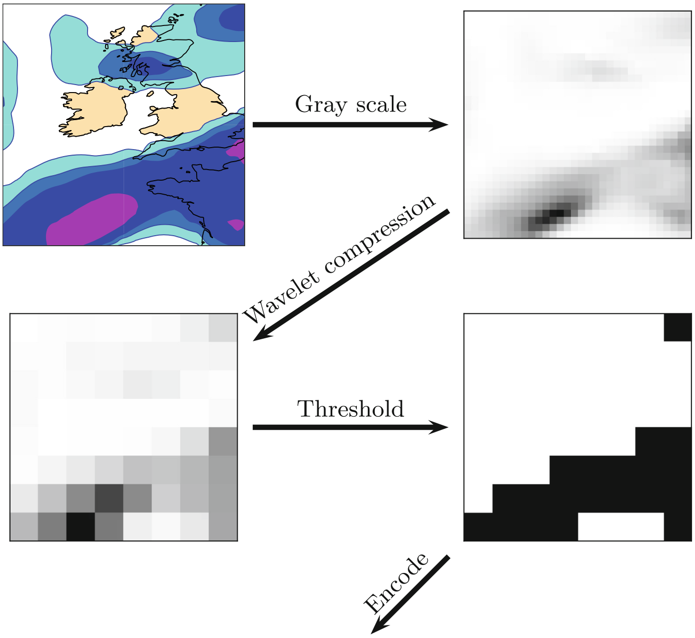
$$F(f) = \langle s, r \rangle$$

where:

- s is a bit vector, representing the shape of f, and
- r is a reference value, capturing the intensity of the field f.

The fingerprinting method proposed is as follows:

1. the meteorological field is considered as a 2D grayscale image;
2. a reference value is selected (for example the mean, or the median of the field);
3. the field is compressed using wavelet compression;
4. the reference value is used as a threshold to convert the compressed image into a bitmap;
5. the bits that make the bitmap are extracted and form the shape part of the fingerprint.



$$F(f) = \langle 00000001 \cdots 11110001, 0.003 \rangle$$

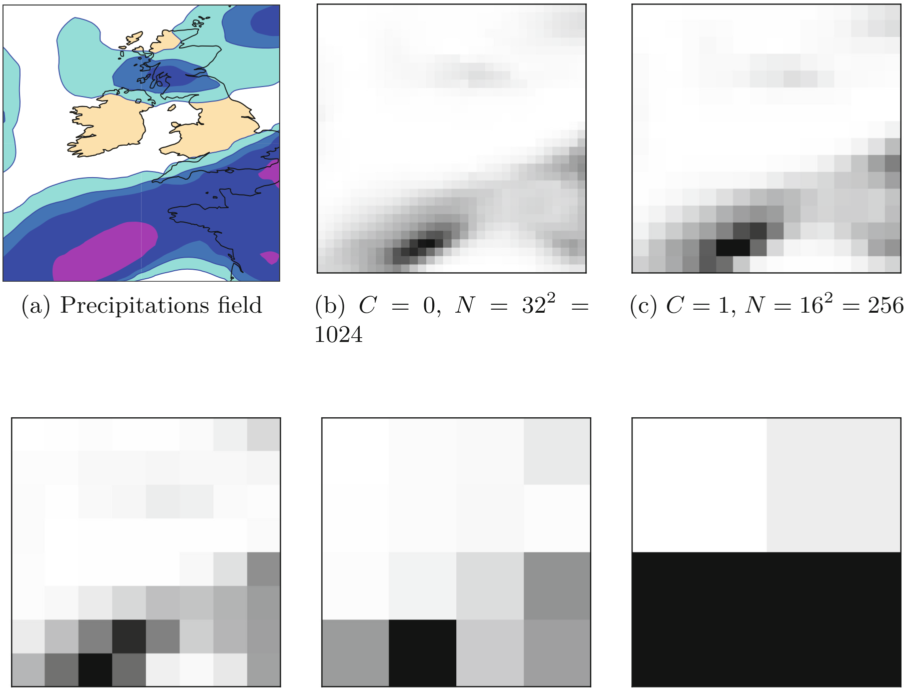
Fig. 2. Algorithm: field fingerprints are computed using wavelet compression and thresholding. In this example, 0.003 is the average value of the field.

The first step is only described here to stress that the algorithm expects the actual values of the field as input, and not a graphical representation (fields are not images). In the case of this research, fields are already available in a binary form, so the first step is not necessary. The method is illustrated in Fig. 2. In that example, the fingerprint is a tuple consisting of a 64 bits vector and a floating-point value. In a modern computer, this would use 128 bits of memory.

4.2 Wavelet Compression

A Discrete Wavelet Transform (DWT) decomposes a signal into approximation and details coefficients; the approximation is a smoothing of the signal, and capture large scale features, while details represent smaller variations around the approximation. The original signal can we reconstructed from all coefficients. Wavelet compression is performed by selecting the approximation coefficient of a given stage of the DWT and discarding the detail coefficients.

We will define the compression factor C as the level of the DWT. As C increases, the number values in the compressed field is divided by 4 (Fig. 3).



(a) Precipitations field

(b) $C = 0$, $N = 32^2 = 1024$

(c) $C = 1$, $N = 16^2 = 256$

(d) $C = 2$, $N = 8^2 = 64$

(e) $C = 3$, $N = 4^2 = 16$

(f) $C = 4$, $N = 2^2 = 4$

Fig. 3. Grey scale images showing the result of wavelet compression of a field of precipitations. C is the compression factor, N is the number of data values remaining after compression.

4.3 Query

Looking up for analogues is done by solving the nearest neighbour problem in a database of fingerprints. In that study, the fingerprints are held in a simple array structure in memory, are they are small enough, and the lookup is implemented as a linear scan. The performance of this setup is sufficient for interactive use. More elaborate data structures and algorithm will be considered at a later stage.

To querying the database for analogues, the user needs to present a meteorological field over a similar area and with the same number of grid points as our current setup. This could be for example today's weather, extracted from the latest analysis from a NWP centre. The fingerprint of the query field is computed and is compared to existing fingerprint. Fingerprints are considered close if the Hamming distance [37] of their bit vectors are close, and their reference values are also close.

4.4 Formal Definition

The problem we are trying to address can be formalised as:

Let v be a meteorological variable (e.g. surface pressure, wind speed...).

Let $\mathcal{A}_v$ be the set of all meteorological fields in the archive for this variable. Assuming that all the fields are defined over the same grid (same geographical coverage, same resolution), $\mathcal{A}_v$ can be considered a subset of $\mathbb{R}^n$, with n being the number of grid points.

Let D be a distance function between the elements of A_v (typically the $L2$-norm).

Let F be the set of fingerprints.

Let δ be a distance function between the elements of F.

We are looking for a mapping $F_v\colon \mathcal{A}_v \mapsto \mathcal{F}$ such that:

$$\begin{aligned} &\forall f_1, f_2, f_3 \in \mathcal{A}_v, D(f_1, f_2) \leq D(f_1, f_3) \\ \iff &\delta(F_v(f_1), F_v(f_2)) \leq \delta(F_v(f_1), F_v(f_3)). \end{aligned} \tag{1}$$

Intuitively, this means that F_v "preserves distances", e.g. if fields are close according to the distance D, their fingerprints must also be close according to the distance δ. Similarly, fields that are far apart must have fingerprints that are far apart. A study of distance preserving embeddings is available from [38].

The aim of this work is to find a mapping that mostly satisfy relation (1), i.e. a mapping for which the relation is true for most elements of $\mathcal{A}_v$.

Traditionally, distance between meteorological fields is computed using the root mean square deviation (RMSD), which is equivalent to the $L2$-norm. Other distances such as Pearson correlation coefficient (PCC) are also used. [39] show the limitations of such metrics. In this study, we will use the $L2$-norm when comparing field, as it is the most commonly used metric in meteorology.

4.5 Validation of the Mapping

As we are considering various fingerprinting schemes, we will compare how "effective" they are. We define the effectiveness of a mapping is a measure of number of elements of $\mathcal{A}_v$ for which relation (1) hold.

A scheme is perfectly effective if for every query q, we always find the field which is closest to q according to the distance D. This can also be stated as: if m be the best match when querying the system with q, the scheme is perfectly effective if there are no field closer to q than m according to the distance D. Conversely, the more fields are closer to q than m, the less effective the method. So, to measure the effectiveness of the fingerprinting scheme, we count how many fields are closer to q than m. Instead of generating dummy query fields, we use every fields from the archive to query a set composed of all other fields.

Using the definitions from Sect. 4.4, for each field q in $\mathcal{A}_v$, let $\mathcal{A}_v^q = \mathcal{A}_v \backslash \{q\}$ be the dataset that excludes this field.

Let m be the best match when querying $\mathcal{A}_v^q$ with q.

Let $\xi_D(q)$ be the query error, defined as the number of fields that are closer to q than m according to a distance D, normalised by the total number of field in $\mathcal{A}_v$:

$$\xi_D(q) = \frac{|\{f \in \mathcal{A}_v^q \mid D(f,q) < D(m,q)\}|}{|\mathcal{A}_v^q|} \ .$$

$\xi_D(q) = 0$ if the result of querying $\mathcal{A}_v^q$ with q returns the closest field to q according to the distance D, and $\xi_D(q) = 1$ if the resulting field is the furthest away according to D.

We consider the scheme to be validated if $\xi_D(q)$ is negligibly small (e.g. less that 0.05, i.e. 5%) for a large number of values of q (e.g. 80%). This means that for 80% of the queries, less than 5% of all the fields in the dataset will considered a better match than the closest field according to D.

4.6 Choice of the Compression Factor C

In order to select a value for the compression factor C, we compute $\xi_{L2}(q)$ for every field q of the dataset. We then consider the percentage of fields of the dataset for which the $\xi_{L2}(q)$ is below a given value.

Figure 4 shows, for two representative meteorological variables, the sorted distribution of the values ξ_{L2} against the queries, for various values of the compression factor C. Figure 4b shows that for $C = 3$ and for 80% of the queries, less than 4% of the fields are actually closer than the best match. Plotting such graphs for all selected meteorological variables shows that the best results are obtained with the compression factor $C = 3$. This can be explained as follows:

For $C = 1$ and $C = 2$, the compressed field retain a lot of detail and the resulting fingerprints retain many dimensions, and we are affected by the curse of dimensionality.

For $C = 4$, too much information is lost, and dissimilar fields are more likely to have similar fingerprints, thus increasing the probability of mismatching results.

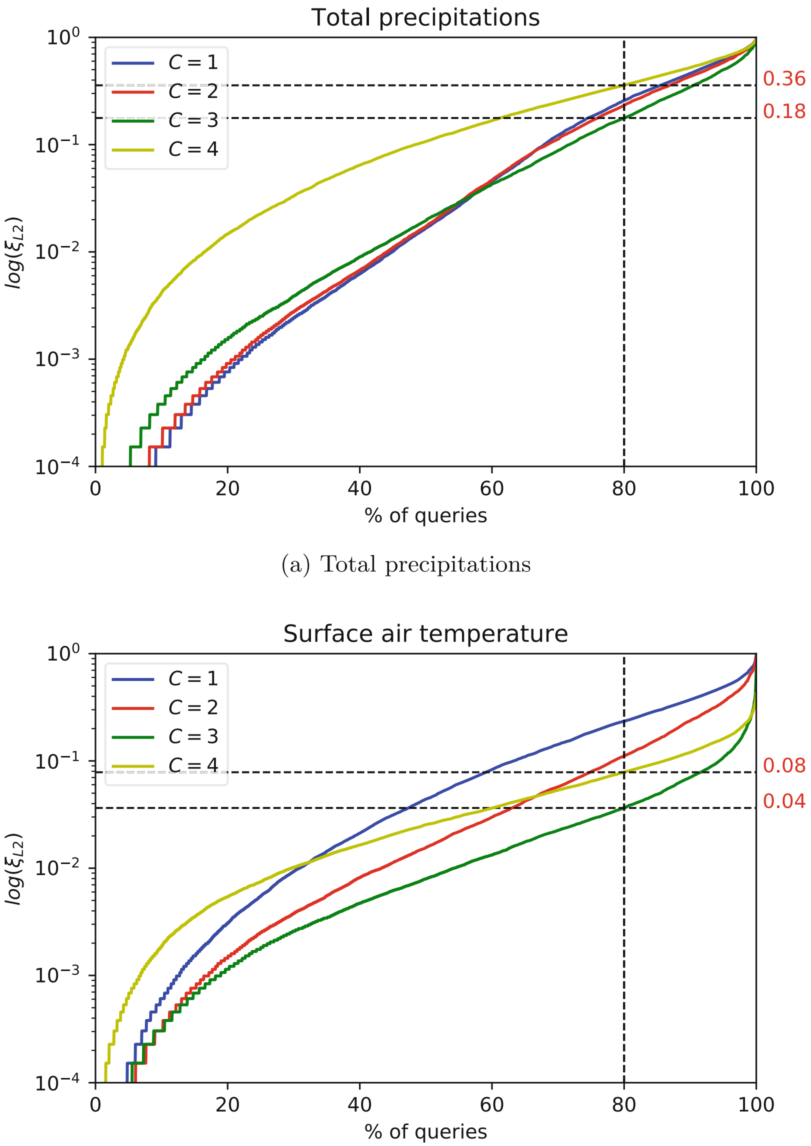
We can see that for total precipitations (Fig. 4a), the results are not as good as for the surface air pressure. This is because this field is not as smooth and continuous, and is by nature not easily captured by the multi-resolution aspect of wavelets.

The value $C = 3$ provides enough information reduction so that generated fingerprints are small, while having a high effectiveness so that matching of fingerprints will provide good results.

4.7 Similarity Measure Between Fingerprints

In Sect. 4.1, we define the fingerprint of f as $F(f) = \langle s, r \rangle$ where:

- s i a bit vector representing the shape of f, and
- r is a reference value, capturing the intensity of the field f.



(a) Total precipitations

(b) Surface air temperature

Fig. 4. Choice of the compression factor C. The plots shown are sorted distributions of ξ_{L2} for various values of C. For *Total precipitation*, we see that for $C = 4$, the value of ξ_{L2} at 80% is 0.36. This means that for 20% of the queries, there are more than 36% of all the fields in the dataset that are considered a better match than the closest field according to $L2$. For $C = 3$, this value drops to 18%. For *Surface air temperature*, we can see that the results are much better, and that for $C = 4$, the value at 80% is 0.08 (8%) and for $C = 3$, the value at 80% is 0.04 (4%). In both cases, $C = 3$ gives the best results.

We use the mean of the field for r. We then define the distance between the fingerprints $\langle s_1, r_1 \rangle$ and $\langle s_2, r_2 \rangle$ as:

$$\delta(\langle s_1, r_1 \rangle, \langle s_2, r_2 \rangle) = \begin{cases} hamming(s_1, s_2) & if s_1 \neq s_2, \\ |r_1 - r_2| & \text{otherwise.} \end{cases}$$

This means that we first compare the shapes, and if they are identical, we then compare the intensities of the two fingerprints (lexical ordering). For this method, we show the best results are for $C = 3$, as in paragraph Sect. 4.6.

This is an interesting result as it shows that a value of $C = 3$ is sufficient for s to capture the shape of the field. In that case, s is 16 bits long. The mean r can easily be encoded using 16 bits, without loss of effectiveness:

$$r_{16bits} = \left\lfloor 2^{16} \frac{(r - min_v)}{(max_v - min_v)} \right\rfloor.$$

Where $\lfloor x \rfloor$ is the nearest smaller integer from x (floor), and min_v and max_v are the minimum and maximum values possible for the meteorological variable v. In this case, the fingerprint can be encoded over 32 bits. Tests using the median instead of the mean do not give better results.

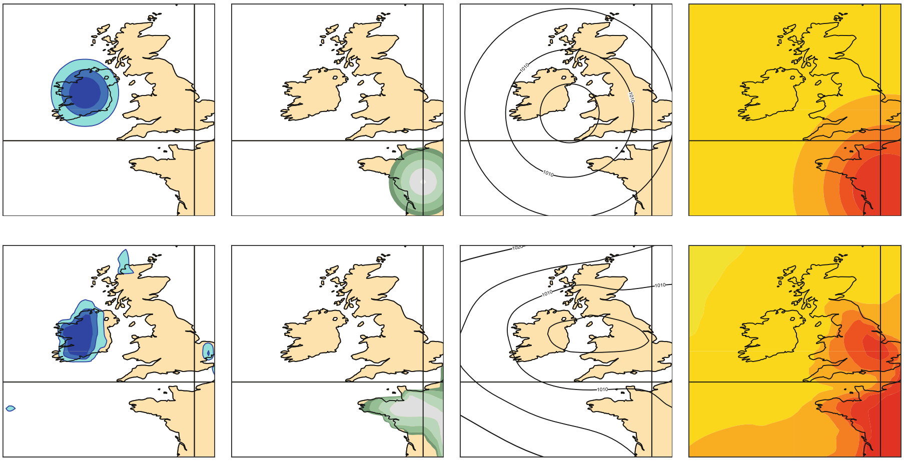
5 Implementation and Results

The code implemented for this work is written in Python, using NumPy [40], SciPy [41], Matplotlib [42], PyWavelet [43]. Bespoke Python module have been developed to interface with ECMWF's GRIB decoder [44], to decode the meteorological fields, as well as ECMWF's plotting package MAGICS [32,45], to plot maps. The various fingerprinting methods, as well as the code to estimate their effectiveness. Experiments are run using Jupyter, previously known as iPython notebook [46].

Several artificial patterns are used to query the system (see Fig. 5). These patterns do not represent realistic meteorological fields. They could nevertheless be the kind of pattern that the user could query:

- Fig. 5a: some heavy precipitations over Ireland only.
- Fig. 5b: some snow in western France.
- Fig. 5c: a system of high pressure over the British Isles.
- Fig. 5d: a heat wave over the south east of England and France.

In each case, the system will return a field from the archive that matches the query provided.



(a) Total precipitations.

(b) Snow cover.

(c) Pressure.

(d) Air temperature.

Fig. 5. Using artificial fields as queries (first row), and the corresponding best matches (second row).

6 Conclusion and Future Work

In this work the first wavelet base retrieval system for weather analogue has been introduced. Results shows that 32 bits fingerprints are sufficient to represent meteorological fields over a 1700 km × s1700 km region, and that distances between fingerprints provide a realistic proxy to the distance between fields. The small size of the fingerprint means that they can be stored in memory, leading to very short lookup time, fast enough to allow for interactive queries.

As part of our future work, will be considering a method that allows users to describe type of weathers in an interactive fashion. Users will be provided with a tool to “draw” the field they are looking. The pattern drawn will be used as a query to the system, and similar fields will be returned. One of the main challenge of this method will be to ensure that the user’s input is realistic from a meteorological point of view.

During our initial research, we have been focussing on weather patterns over the British Isles. As part of the future work, we will consider extending the system to the whole globe.

Weather situations are really similar if all of the parameters (temperature, pressure, wind, etc.) are also similar. We will study how the fingerprinting scheme implemented so far can be extended so that it takes into account several parameters and what are the implication on the index and the matching algorithms.

References

1. Delle Monache, L., Eckel, F.A., Rife, D.L., Nagarajan, B., Searight, K.: Probabilistic Weather Prediction with an Analog Ensemble. Mon. Wea. Rev. **141**(10), 3498–3516 (2013)
2. Van den Dool, H.: A new look at weather forecasting through analogues. Mon. Weather Rev. **117**(10), 2230–2247 (1989)
3. Van den Dool, H.: Searching for analogues, how long must we wait? Tellus A **46**(3), 314–324 (1993)
4. Zorita, E., von Storch, H.: The analog method as a simple statistical downscaling technique: comparison with more complicated methods, pp. 1–16, August 1999
5. Evans, M., Murphy, R.: A historical-analog-based severe weather checklist for central New York and northeast Pennsylvania, pp. 1–8, February 2013
6. Sanderson, M.G., Hanlon, H.M., Palin, E.J., Quinn, A.D., Clark, R.T.: Analogues for the railway network of Great Britain. Meteorol. Appl. **23**(4), 731–741 (2016)
7. Jacobs, C.E., Finkelstein, A., Salesin, D.H.: Fast multiresolution image querying. In: Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques, pp. 277–286. ACM (1995)
8. Baluja, S., Covell, M.: Waveprint: efficient wavelet-based audio fingerprinting. Pattern Recogn. **41**(11), 3467–3480 (2008)
9. Orio, N.: Music Retrieval: A Tutorial and Review. Now Publishers Inc., Boston (2006)
10. Veltkamp, R., Burkhardt, H., Kriegel, H.P.: State-of-the-Art in Content-Based Image and Video Retrieval. Springer Science & Business Media, Dordrecht (2013). https://doi.org/10.1007/978-94-015-9664-0
11. Daubechies, I.: Orthonormal bases of compactly supported wavelets. Commun. Pure Appl. Math. **41**(7), 909–996 (1988)
12. Walker, J.S.: A primer on wavelets and their scientific applications, pp. 1–156, June 2005
13. Stollnitz, E.J., DeRose, T.D., Salesin, D.H.: Wavelets for computer graphics: a primer part 1, pp. 1–8 (1995)
14. Stollnitz, E.J., DeRose, T.D., Salesin, D.H.: Wavelets for computer graphics: a primer part 2, pp. 1–9 (1995)
15. Stollnitz, E.J., DeRose, T., Salesin, D.H.: Wavelets for Computer Graphics - Theory and Applications. Morgan Kaufmann, San Francisco (1996)
16. Balan, V., Condea, C.: Wavelets and Image Compression. Telecommunication Standardization Sector of lTU, Leden (2003)
17. Porwik, P., Lisowska, A.: The Haar-wavelet transform in digital image processing: its status and achievements. Mach. Graph. Vision **13**(1/2), 79–98 (2004)
18. Shapiro, J.M.: Embedded image coding using zerotrees of wavelet coefficients. IEEE Trans. Signal Process. **41**(12), 3445–3462 (1993)
19. Walker, J.S., Nguyen, T.Q.: Wavelet-based image compression. In: Rao, K.R. et al.: The Transform and Data Compression Handbook. CRC Press LLC, Boca Raton (2001)
20. Zeng, L., Jansen, C., Unser, M., Hunziker, P.: Extension of wavelet compression algorithms to 3D and 4D image data: exploitation of data coherence in higher dimensions allows very high compression ratios, pp. 1–7, October 2011
21. Patrikalakis, N.M.: Wavelet based similarity measurement algorithm for seafloor morphology. Massachusetts Institute of Technology (2006)

22. Regentova, E., Latifi, S., Deng, S.: A wavelet-based technique for image similarity estimation. In: ITCC-00, pp. 207–212. IEEE (2000)
23. Pauly, O., Padoy, N., Poppert, H., Esposito, L., Navab, N.: Wavelet energy map: a robust support for multi-modal registration of medical images. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2009, pp. 2184–2191. IEEE (2009)
24. Traina, A.J.M., Castañón, C.A.B., Traina, Jr., C.: MultiWaveMed: a system for medical image retrieval through wavelets transformations. In: IEEE Computer Society, June 2003
25. Marsolo, K., Parthasarathy, S., Ramamohanarao, K.: Structure-based querying of proteins using wavelets. In: Proceedings of the 15th ACM International Conference on Information and Knowledge Management, pp. 24–33. ACM (2006)
26. Cattani, C., Ciancio, A.: Wavelet clustering in time series analysis. Balkan J. Geom. Appl. **10**(2), 33 (2005)
27. Kocaman, Ç., Özdemir, M.: Comparison of statistical methods and wavelet energy coefficients for determining two common PQ disturbances: sag and swell. In: International Conference on Electrical and Electronics Engineering, ELECO 2009, pp. I-80–I-84. IEEE (2009)
28. Phuc, N.H., Khanh, T.Q., Bon, N.N.: Discrete wavelets transform technique application in identification of power quality disturbances (2005)
29. Gomez-Glez, J.F.: Wavelet methods for time series analysis, pp. 1–45, February 2009
30. Popivanov, I., Miller, R.J.: Similarity search over time-series data using wavelets. In: 18th International Conference on Data Engineering, Proceedings, pp. 212–221. IEEE (2002)
31. Raoult, B.: Architecture of the new MARS server. In: Sixth Workshop on Meteorological Operational Systems, ECMWF, 17–21 November 1997, Shinfield Park, Reading, pp. 90–100 (1997)
32. Woods, A.: Archives and graphics: towards MARS, MAGICS and Metview. In: The European Approach, Medium-Range Weather Prediction, pp. 183–193 (2006)
33. Frauenfeld, O.W., Zhang, T., Serreze, M.C.: Climate change and variability using European Centre for Medium-Range Weather Forecasts reanalysis (ERA-40) temperatures on the Tibetan Plateau. J. Geophys. Res. Atmos. (1984–2012) **110**(D2) (2005)
34. Santer, B.D., Wigley, T.M., Simmons, A.J., Kållberg, P.W., Kelly, G.A., Uppala, S.M., Ammann, C., Boyle, J.S., Brüggemann, W., Doutriaux, C.: Identification of anthropogenic climate change using a second-generation reanalysis. J. Geophys. Res. Atmos. (1984–2012) **109**(D21) (2004)
35. Dee, D., Uppala, S., Simmons, A., Berrisford, P., Poli, P., Kobayashi, S., Andrae, U., Balmaseda, M., Balsamo, G., Bauer, P.: The ERA-Interim reanalysis: configuration and performance of the data assimilation system. Q. J. Royal Meteorol. Soc. **137**(656), 553–597 (2011)
36. Dee, D., Balmaseda, M., Balsamo, G., Engelen, R., Simmons, A., Thépaut, J.N.: Toward a consistent reanalysis of the climate system. Bull. Am. Meteorol. Soc. **95**(8), 1235–1248 (2014)
37. Sixta, S.: Hamming cube and other stuff, pp. 1–18, May 2014
38. Indyk, P., Naor, A.: Nearest-neighbor-preserving embeddings. ACM Trans. Algorithms (TALG) **3**(3), 31 (2007)
39. Mo, R., Ye, C., Whitfield, P.H.: Application potential of four nontraditional similarity metrics in hydrometeorology. J. Hydrometeorology **15**(5), 1862–1880 (2015)

40. Van Der Walt, S., Colbert, S.C., Varoquaux, G.: The NumPy array: a structure for efficient numerical computation. Comput. Sci. Eng. **13**(2), 22–30 (2011)
41. Jones, E., Oliphant, T., Peterson, P.: SciPy: open source scientific tools for Python (2014)
42. Hunter, J.D.: Matplotlib: a 2D graphics environment. Comput. Sci. Eng. **9**(3), 90–95 (2007)
43. Wasilewski, F.: PyWavelets: discrete wavelet transform in python (2010)
44. Fucile, E., Codorean, C.: GRIB API. A database driven decoding library. In: Twelfth Workshop on Meteorological Operational Systems, ECMWF, 2–6 November 2009, Shinfield Park, Reading, pp. 46–47 (2009)
45. O'Sullivan, P.: MAGICS - the ECMWF graphics package. ECMWF Newslett. (62) (1993)
46. Pérez, F., Granger, B.E.: IPython: a system for interactive scientific computing. Comput. Sci. Eng. **9**(3), 21–29 (2007)