

Distanța Pompeiu-Hausdorff: aplicații în informatică

Albeanu Grigore – Universitatea „Spiru Haret”, București
g.albeanu.mi@spiruharet.ro

Abstract

Distanța Pompeiu-Hausdorff și-a dovedit utilitatea în topologie, analiză matematică, geometrie și în domenii aplicative unde informatica, prin algoritmi eficienți, intervine pentru accelerarea calculului și rezolvarea, în timp real a recunoașterii formelor, urmării obiectelor etc. Lucrarea prezintă aplicații ale distanței Pompeiu-Hausdorff în grafica pe calculator (calitatea reprezentărilor spațiale cu ajutorul poligoanelor) și analiza imaginilor (identificarea unor forme în imagini date). Sunt descrise extensii ale metricii standard și algoritmi de calcul.

1. Introducere

Compararea imaginilor în general și a textelor imagine, în particular, recunoașterea formelor, precum și monitorizarea obiectelor necesită algoritmi eficienți. O clasă largă de algoritmi utilizează distanța Pompeiu-Hausdorff a cărei aplicabilitate a fost demonstrată prin numeroase studii și cercetări publicate în reviste științifice aparținând fluxului principal de publicații. Prezenta lucrare analizează, din punct de vedere informatic, o parte din aceste aspecte.

2. Distanța Pompeiu-Hausdorff

Cei mai mulți matematicieni utilizează denumirea de „distanță Hausdorff” pentru a exprima distanța dintre două mulțimi. Introdusă inițial de către Pompeiu în teza de doctorat ([4], [6], [9], [17], [18]) prin

$$D_p(A, B) = \frac{1}{2} \left(\sup_{x \in A} \inf_{y \in B} d(x, y) + \sup_{x \in B} \inf_{y \in A} d(x, y) \right),$$

(unde d este o metrică între puncte ale spațiului metric considerat) aceasta a fost studiată și de Felix Hausdorff, care în 1914, citând lucrarea lui Pompeiu, introduce distanța dintre două mulțimi prin intermediul operației \max [6]:

$$D_{PH}(A, B) = \max \left\{ \sup_{x \in A} \inf_{y \in B} d(x, y), \sup_{x \in B} \inf_{y \in A} d(x, y) \right\}.$$

Ambele formule au suportat generalizări, studii și au fost aplicate în diverse domenii ([4], [7], [11], [13], [19] etc.). De exemplu, Aseev, Tetenov și Maksimova [4], definesc distanța generalizată Pompeiu astfel:

$$D_p^\alpha(A, B) = \frac{1}{2^{1/\alpha}} \left[\left(\sup_{x \in A} \inf_{y \in B} d(x, y) \right)^\alpha + \left(\sup_{x \in B} \inf_{y \in A} d(x, y) \right)^\alpha \right]^{1/\alpha},$$

pentru $1 < \alpha < \infty$, unde A și B sunt submulțimi ale axei reale, iar $d(x, y) = |x - y|$.

Huttenlocher, Klanderman și Rucklidge [14] au propus mai multe extensii ale distanței D_{PH} cu aplicabilitate în analiza imaginilor pe baza distanței Euclidiene $d(x, y) = \|x - y\|$ și $d^*(a, X) = \min_{x \in X} d(a, x)$ bazate pe ierarhizare. Le vom unifica prin:

$$d_{PH}^{\alpha}(A, B) = {}^{\alpha}K_{a \in A}^{th} d^*(a, B),$$

where $\alpha \in [0, 100]$, iar operatorul ${}^{\alpha}K_{a \in A}^{th}$ reprezintă a K-a valoare a distanței calculate astfel încât $K/\text{card}(A) = \alpha/100$. Dacă $\alpha = 50$ atunci se obține mediana. Autorii aplică D_{PH} modificat cu $\alpha \in \{50, 75, 90\}$. Dubuisson & Jain (1994), în contextul analizei similarității obiectelor [11], au propus utilizarea formulei d_{PHM} : $d_{PHM}(A, B) = \max(d(A, B), d(B, A))$,

$$\text{unde } d(A, B) = \frac{1}{\text{card}(A)} \sum_{a \in A} d^*(a, B).$$

Experimental, a fost dovedit că: 1) valorile d_{PHM} dintre imagini cresc odată cu distanța dintre muchii și 2) este robustă la prezența eventualelor puncte care apar din erori de segmentare. Pentru autorii menționați [11, 14], mulțimile A și B sunt finite.

Wang și Tan, în [25], consideră și vecinii punctelor care servesc la aflarea distanței, folosind medierea și caracteristicile obiectelor analizate.

3. Aplicații în informatică

Distanta de la un punct x la un poligon A se definește astfel: este distanța euclidiană de la x la cel mai apropiat punct y aparținând poligonului A.

Fie A și B două mulțimi. Cele două componente ale distanței Pompeiu-Hausdorff sunt $d_{A \rightarrow B}$ (distanța dintre A și B) și $d_{B \rightarrow A}$ (distanța dintre B și A). Dacă A și B sunt poliliniile din plan (Figura 1; A are 3 segmente, $x \in A$; B are 2 segmente, $y \in B$) se vede că cele două componente pot avea valori diferite [12].

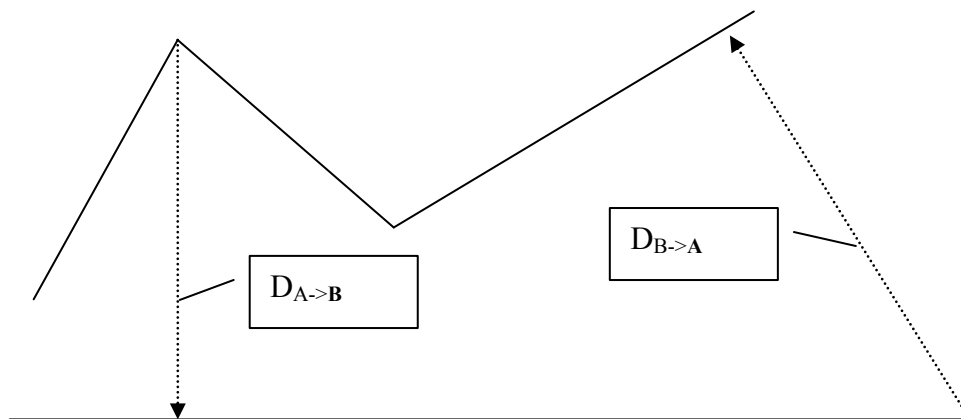


Figure 1. The two components of Pompeiu-Hausdorff distance

În [12], Hangouët (1995) ilustrează următoarele proprietăți: 1) cele două componente, de obicei, au valori diferite; 2) Vectorul care indică distanță de la un obiect la celălalt este perpendicular pe al doilea obiect (la utilizarea distanței euclidiene ca distanță dintre puncte în plan); 3) distanța Pompeiu-Hausdorff este influențată din punct de vedere al stabilității de excentricitățile poliliniilor; 4) Nu neapărat distanța Pompeiu-Hausdorff este dată prin intermediul vârfurilor poliliniilor, ci poate să rezulte din puncte din interiorul segmentelor.

Un algoritm pentru calculul componenteii $D_{A \rightarrow B}$, conform [12], are următorii pași:

1. Se calculează distanțele de la vârfurile poliliniiei A la poliliniia B.
2. Se verifică dacă nu există puncte ale lui A pentru care să se obțină o distanță mai mare.
3. Se calculează cea mai mare distanță.

Belogay, Cabrelli, Molter și Shonkwiler (1997), în [8], propun un algoritm pentru calculul distanței Pompeiu-Hausdorff dintre două curbe situate în plan, într-o rețea de $N \times N$ celule, unde N este o putere a lui 2. În contextul imaginilor binare, două obiecte din imagine coincid dacă și numai dacă distanța Pompeiu-Hausdorff dintre ele este zero.

În cele ce urmează vom analiza diferite tipuri de obiecte supuse calculului distanței Pompeiu-Hausdorff: poligoane, curbe Bezier/Spline, imagini, scrisul de mână etc.

3.1. Distanța Pompeiu-Hausdorff dintre poligoane convexe

Fie A și B două poligoane, A are n vârfuri, iar B are m vârfuri. Presupunem că acestea nu se intersectează (deci și că unul nu este parte a celuilalt). Fie $x \in A$ și $y \in B$ astfel încât $d(x,y) = D_{PH}(A,B)$. Atunci [5]: a) perpendiculara pe xy în x este dreapta suport a unui segment al lui A ; b) perpendiculara pe xy în y este dreapta suport a unui segment al lui B .

De asemenea, întotdeauna există un vârf $x \in A$ astfel încât $d(x,B) = D_{PH}(A, B)$.

Fie v_0, v_1, \dots, v_{n-1} vârfurile poligonului A și w_0, w_1, \dots, w_{m-1} vârfurile poligonului B , orientate, în ambele cazuri, în sensul invers acelor de ceasornic. Prin metoda naivă, complexitatea obținerii celei mai mari valori d_i necesită $O(mn)$ operații.

Fie d_i distanța de la v_i la B (calculul fiecărei valori d_i implică o complexitate $O(m)$). Fie $y_i \in B$ (fie vârf, fie piciorul perpendicularei din v_i pe B) astfel încât $d_i = d(v_i, y_i)$. Notăm cu σ secvența $y_0, y_1, \dots, y_{n-1}, y_0$. Spunem că σ merge în sens invers acelor de ceasornic în y_i dacă v_{i+1} este la stânga lui $y_i v_i$, altfel spunem că mișcarea este în sensul acelor de ceasornic (v_{i+1} este la dreapta lui $y_i v_i$), respectiv σ este staționară dacă v_{i+1} aparține segmentului $y_i v_i$. Schimbarea sensului mișcării este posibilă. Mai precis: σ își schimbă direcția exact de 2 ori.

Următorul algoritm [5] efectuează calculul cu $O(m+n)$ operații.

1. Calculează d_0 și deci află și y_0 . $D(A, B) \leq d_0$, $k \leq 0$;
2. Dacă v_{k+1} este la stânga lui $y_k v_k$ atunci scanăm poligonul B în sens invers acelor de ceasornic, pornind din y_k , până când: a) există o muchie a lui B , notată $w_q w_{q+1}$, astfel încât piciorul perpendicularei din v_{k+1} este z situat pe $w_q w_{q+1}$, între w_q și w_{q+1} . Atunci $y_{k+1} \leq z$; b) există un vârf al lui B , notat w_q , astfel încât perpendiculara pe $v_{k+1} w_q$ în w_q este o dreapta suport a lui B , atunci $y_{k+1} \leq w_q$.
3. Dacă v_{k+1} este la dreapta lui $y_k v_k$ se aplică raționamentul de la 2) obținând y_{k+1} prin scanarea lui B , pornind din y_k , dar în sensul acelor de ceasornic.
4. Dacă v_{k+1} este pe $y_k v_k$, dar nu neapărat între y_k și v_k atunci $y_{k+1} \leq y_k$.
5. $D(A, B) = \max \{D(A, B), d(v_{k+1}, y_{k+1})\}$; $k \leq k \bmod n$; Dacă $k \neq 0$ se continuă cu pasul 2.

Pentru a calcula $d_{HP}(A, B)$, algoritmul de mai sus se utilizează de două ori: odată pe calculul $D(A, B)$, apoi pentru calculul $D(B, A)$, iar în final se ia cea mai mare valoare.

3.2. Distanța Pompeiu-Hausdorff dintre curbe de aproximare

În proiectarea asistată de calculator și modelarea geometrică a curbilor cu ajutorul unor poligoane de control, curbele de aproximare Bezier, respectiv Spline joacă un rol esențial. Acestea sunt utile și în aplicații de deformare a formelor (aplicații de morfologie).

Pentru a calcula distanța Pompeiu-Hausdorff dintre două curbe de aproximare, definite prin poligoane de control, sunt necesare proceduri specifice precum: calculul proiecției unui punct de pe curba A pe curba B; determinarea rădăcinilor unei ecuații polinomiale; analiza cazurilor posibile.

Complexitatea algoritmului este ridicată datorită metodelor numerice care intervin. Pentru situații urgente se poate recurge la discretizarea curbelor și aplicarea unui algoritm precum cel descris în [8]. O descriere completă a algoritmilor de calcul a distanței Pompeiu-Hausdorff va fi prezentată în altă lucrare.

3.3. Distanța Pompeiu-Hausdorff dintre imagini

Imaginile joacă un rol important în foarte multe aplicații ale lumii moderne. De la identificarea obiectelor până la recunoașterea feței, urmărirea obiectelor în filme, sau prin intermediul camerelor de supraveghere sau radare, calculul distanței Pompeiu-Hausdorff intervine direct sau indirect.

După cum a fost menționat mai sus, Huttenlocher, Klanderman și Rucklidge, în [14] au definit și experimentat utilizarea distanței Hausdorff și a unor extensii ale acesteia pentru compararea imaginilor ca mulțimi de pixeli.

În [2], respectiv [3], analiza textului este realizată prin studiul imaginilor binare ale textului. Prin extensie ideile prezentate au aplicabilitate în recunoașterea textului „pictat”, respectiv pentru căutarea cunoștințelor în imagini.

Detectarea și recunoașterea feței în imagini și cadre video este posibilă prin noile tehnici bazate de distanța Pompeiu-Hausdorff modificată pentru a lua în considerare atribute (caracteristici), modelate crisp [1, 15, 23, 25] sau fuzzy [13].

Vignon, Lovell & Andrews (20012), în [24], au utilizat , definit în [14], pentru urmărirea unui obiect deplasabil într-o scenă. Se utilizează o imagine model A și transformări t aplicate modelului B pentru a vedea gradul de potrivire dintre A și $t(B)$, unde t reprezintă transformări geometrice de tip scalare, translație, respectiv rotații. Monitorizarea este realizată prin calculul distanței Pompeiu-Hausdorff (standard sau modificată) $d_{PH}(A, t(B))$.

4. Concluzii

Distanța Pompeiu-Hausdorff și-a dovedit utilitatea în topologie, analiză matematică, geometrie și în domenii aplicative unde informatica, prin algoritmi eficienți intervine pentru accelerarea calculului și rezolvarea, în timp real a recunoașterii formelor, urmării obiectelor ș.a. Este meritul lui Pompeiu care a contribuit la definirea acestei distanțe și a cercetătorilor care au aplicat ideile lui Pompeiu și Hausdorff pentru a rezolva multiple probleme ale științei și tehnologiei.

5. Bibliografie

- [1]. J.L. Alba, A. Pujol, A. Lopez, and J.J. Villanueva, "Improving Shape-Based Face Recognition by Means of a Supervised Discriminant Hausdorff Distance", *Proceedings of the International Conference on Image Processing*, 2003, pp.901-904.
- [2]. A. Andreev, N. Kirov, "Some Variants of Hausdorff Distance for Word Matching", *Review of the National Center for Digitization*, 12, 2008, pp3-8.
- [3]. A. Andreev, N. Kirov, "Hausdorff distances for searching in binary text images", *Serdica Journal of Computing*, 3 (1), 2009, pp23-46.
- [4]. V.V. Aseev, A.V. Tetenov & A.P. Maksimova, "The Generalized Pompeiu Metric in the Isometry Problem for Hyperspaces", *Mathematical Notes*, vol. 78, no. 2, pp149-155, 2005.

- [5]. M.J. Atallah, "A Linear Time Algorithm for the Hausdorff Distance Between Convex Polygons", *Computer Science Technical Reports*, Paper 363, 1983, <http://docs.lib.purdue.edu/cstech/363>.
- [6]. T. Birsan, D. Tiba, "One Hundred Years since the Introduction of the Set Distance by Dimitrie Pompeiu", *System Modeling and Optimization*. IFIP International Federation for Information Processing V. 199, 2006, pp35-39.
- [7]. E. Baudrier, G. Millon, F. Nicolier, S. Ruan, "A new similarity measure using Hausdorff distance map", *IEEE Int. Conf. on Image Processing*, 2004, pp669-672.
- [8]. E. Belogay, C. Cabrelli, U. Molter, R. Shonkwiler, "Calculating the Hausdorff Distance Between Curves", *Information Processing Letters*, 64, 1997, pp17-22.
- [9]. T. Birsan, D. Tiba, Un secol de la publicarea tezei de doctorat a lui Dimitrie Pompeiu, "*Recreații matematice*", 2, 2005, pp85-89.
- [10]. R. Dorrigiv, S. Durocher, A. Farzan, R. Fraser, Al. Lopez-Ortiz, J.I. Munro, Al. Salinger, and M. Skala, "Finding a Hausdorff Core of a Polygon: On Convex Polygon Containment with Bounded Hausdorff Distance", 11th International Workshop on Algorithms and Data Structures (WADS 2009), *Lecture Notes in Computer Science 5664*, Springer, 2009, pp218-229; The Hausdorff Core Problem on Simple Polygons, 2013, online: http://www.cs.umanitoba.ca/~fraser/papers/haus_core_journal.pdf.
- [11]. M.-P. Dubuisson, A.K. Jain, "A Modified Hausdorff Distance for Object Matching", Proc. International Conference on Pattern Recognition, 1994, pp 566–568.
- [12]. J.F. Hangouët, "Computation of the Hausdorff Distance Between Plane Vector Polylines", *ACSMâSPRS Annual Convention & Exposition Technical Papers*, Bethesda: ACSMâSPRS, 1995, 4, pp1-10.
- [13]. W.-L. Hunga, M.-S. Yang, "Similarity measures of intuitionistic fuzzy sets based on Hausdorff distance", *Pattern Recognition Letters* 25, Elsevier, 2004, pp1603–1611.
- [14]. D.P. Huttenlocher, G.A. Klanderman, W.A. Rucklidge, "Comparing Images Using the Hausdorff Distance", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(9), 1993, pp850-863.
- [15]. J.Joglekar, S.S. Gedam, "Area Based Stereo Image Matching Technique using Hausdorff Distance and Texture Analysis", In: Stilla U et al (Eds) PIA11. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences* 38 (3/W22), 2011, pp109-114.
- [16]. K. Onishi, M. Hoshi, "Computing a Sequence of Circumscribing Polygons for Convex Polygon", *Kyoto University Research Institute for Mathematical Sciences*, <http://www.kurims.kyoto-u.ac.jp/~kyodo/kokyuroku/contents/pdf/1641-09.pdf>, 2009, pp 90-98.
- [17]. D. Pompeiu, D., Sur la continuité de la fonction de variables complexes (Thèse), Gauthier-Villars, Paris, 1905 and Ann. Fac. Sci. de Toulouse, 7, pp 264-315, 1905.
- [18]. D. Pompeiu, "*Opera matematică*", Ed. Acad. Române, București, 1959.
- [19]. A. Pujol, J.J. Villanueva, and J.L. Alba, "A supervised modification of the Hausdorff distance for visual shape classification", *International Journal of Pattern Recognition and Artificial Intelligence*, Vol. 16, No. 3, 2002, pp 349–359.
- [20]. W.J. Rucklidge, "Efficiently Locating Objects Using the Hausdorff Distance", *International Journal of Computer Vision* 24(3), Kluwer Academic Publishers, 1997, pp 251–270.
- [21]. M.I. Schlesinger, E.V. Vodolazskiy, V. Yakovenko, "Strengthened Hausdorff distance between polygons", 2012, pp 1-21.
- [22]. J. Serra, "Hausdorff distances and interpolations", *Proceedings of the fourth international symposium on Mathematical morphology and its applications to image and signal processing*, (H.J.A.M. Heijmans and J.B.T.M. Roerdink, eds.), Kluwer Academic Publishers, 1998, pp 107-114.
- [23]. H. Tan, Y.-J. Zhang, "Computing EigenFace From Edge Images for Face Recognition Based on Hausdorff Distance", *Proceedings of the Fourth International Conference on Image and Graphics*, 2007, pp 639-644.
- [24]. D. Vignon, B. C. Lovell, and R. J. Andrews, "General Purpose Real-Time Object Tracking Using Hausdorff Transforms", *IPMU2002: Special Session on Intelligent Systems for Video Processing*, 2002, pp487-492.
- [25]. J. Wang, Y. Tan, "Hausdorff Distance with k-Nearest Neighbors", ICSI 2012, Part II, LNCS 7332, Springer-Verlag Berlin Heidelberg, 2012, pp 272–281.