Algoritmos de aprendizagem computacional para a modularização de projetos de arquitetura

https://doi.org/10.21814/uminho.ed.32.5

Sofia Feist[1], **Luís Sanhudo**[1], **Vítor Esteves**[1], **Miguel Pires**[2], **António Aguiar Costa**[1]

[1] *BUILT CoLAB – Collaborative Laboratory for the Future Built Environment, Porto*

[2] *CASAIS – Engenharia e Construção, Braga*

Resumo

Apesar da construção modular permitir aumentar significativamente a produtividade na construção, a sua adoção permanece lenta devido ao elevado grau de repetição e normalização necessário para a sua implementação, tipicamente pouco implementado pelos projetistas e requisitado pelos clientes. Para abordar este problema, o presente artigo propõe uma metodologia de modularização a ser implementada após a fase de design estar concluída, aproveitando as vantagens da construção modular enquanto mantem a visão e o design original do edifício. Esta metodologia é baseada numa metodologia de *semi-supervised clustering* para agrupar divisões individuais em grupos conforme a sua semelhança, identificando, posteriormente, um módulo base para representar cada *cluster*. A metodologia proposta é aplicada num caso de estudo no qual foi realizado previamente um processo de modularização manual, servindo como ponto de referência e comparação. Os resultados mostram uma redução de 99,6% do tempo necessário para executar o processo de modularização, justificando o desenvolvimento continuado desta metodologia em trabalhos futuros.

1. Introdução

Analisando a situação atual de produtividade no setor da Arquitetura, Engenharia e Construção (AEC), é possível verificar um atraso considerável em relação a outros setores [1]. Por forma a combater este problema, a indústria AEC tem progressivamente adotado técnicas modulares, de prefabricação e construção *offsite*, em alternativa aos tradicionais métodos de construção *onsite* [2]. Os benefícios da pré-fabricação têm sido amplamente documentados na literatura, e incluem processos de construção mais rápidos e produtivos, resultando em edifícios de maior qualidade [2, 4]. Bertram et al. [5] reportam reduções de 20-50% na duração de projetos modulares e mais de 20% em custos de construção. Por sua vez, a repetição e uniformização da construção modular potencia a reciclagem e reutilização dos materiais, resultando em projetos mais sustentáveis e com menor desperdício [4].

Apesar destas vantagens, a construção modular continua a ser um método pouco popular junto dos clientes devido ao elevado grau de normalização e repetição exigido para reduzir os custos de fabricação [6]. Uma solução para este problema passa por adaptar as estratégias da construção modular ao design existente, em vez de adaptar o design à construção modular. À inversão deste processo denominamos modularização, termo usado para definir o processo de racionalizar o edifício em unidades normalizadas que permitam a sua pré-fabricação *offsite*, sem comprometer a essência do design original. Esta técnica é geralmente impulsionada pelo empreiteiro que beneficia deste aumento de produtividade.

No entanto, esta entidade é normalmente envolvida no empreendimento de construção após o fim do processo de design [3], numa fase onde a sua capacidade de influenciar o projeto é limitada, constituindo um obstáculo à implementação destas técnicas modulares. Notavelmente, a modularização pós-design de projetos não modulares pode envolver um processo de racionalização complexo e moroso, originando atrasos e custos acrescidos de projeto que podem ser evitados introduzindo automatização no processo de modularização.

Para abordar este problema, o presente artigo propõe desenvolver e validar em caso de estudo uma metodologia baseada em *machine learning* para a modularização automática de projetos não-modulares. Esta metodologia usa um algoritmo de *semi-supervised clustering* para agrupar divisões com características semelhantes em *clusters*, usando feedback do utilizador sobre os resultados para guiar o algoritmo e obter resultados mais precisos. A partir destes resultados, é possível a identificação de um módulo base para representar cada conjunto de divisões semelhantes. A metodologia proposta pode ser aplicada durante a fase da pré-construção, aproveitando os benefícios da construção modular e, simultaneamente, mantendo a visão e o design original do edifício.

2. Estado da Arte

No setor AEC, o processo de modularização é geralmente heurístico, sendo que os módulos são identificados com base nas especificações de cada projeto, derivado da experiência do arquiteto ou engenheiro [7]. Este processo não é, no entanto, de fácil implementação, uma vez que cada projeto de arquitetura é único e requer soluções personalizadas, resultando num processo de modularização subjetivo e dependente de múltiplas variáveis como a funcionalidade do módulo, os limites associados ao processo de pré-fabricação, bem como os respetivos custos. Ao contrário de outras indústrias [7], são poucos os estudos deste processo na indústria AEC.

Isaac et al. [8] desenvolveram uma metodologia de *hierarchical clustering* baseada em gráficos para decompor o projeto de um edifício em módulos de acordo com as suas taxas de substituição relativa. De forma semelhante, Samarasinghea et al. [7] aplica uma metodologia de *hierarchical clustering*, combinando-a com uma *Dependency Structure Matrix* para otimização do número de módulos, tendo em conta os custos de montagem e transporte.

Noutro estudo, com o objetivo de diminuir o número de componentes únicos de um edifício, Mohamad et al. [9] propõem um método de modularização de projetos pontuais baseado na subdivisão de um edifício em grelha e no alinhamento e normalização dos componentes desse mesmo edifício com a grelha criada. Por fim, Tserng et al. [10] foca a modularização de sistemas mecânicos, elétricos e hidráulicos, aplicando uma técnica de processamento baseada em regras para decompor estes sistemas em múltiplas componentes.

Dos estudos mencionados acima, apenas Isaac et al. [8] apresentam um procedimento sistemático e automatizado para a modularização de projetos de construção. No entanto, este estudo foca-se na modularização da interface entre módulos e não nos próprios módulos. A metodologia apresentada neste artigo pretende abordar esta lacuna.

3. Metodologia

Para promover a modularização de projetos não modulares, foi desenvolvida uma metodologia baseada em *semi-supervised clustering*, que usa informação sobre as divisões de um edifício para identificar módulos que possam ser industrializados para a construção. Como ilustrado na Figura 1, esta metodologia recebe como *input* um modelo BIM previamente preparado, extraindo e processando a informação do modelo para alimentar o algoritmo de *semi-supervised clustering*. O algoritmo calcula a semelhança entre divisões com base na informação extraída e agrupa-as em *clusters* distintos. Posteriormente, as divisões com maior grau de incerteza durante o *clustering* são manualmente validados pelo utilizador, que identifica as atribuições corretas e incorretas dos módulos. Este *feedback* é depois utilizado numa nova iteração da análise de *clustering*. O processo é repetido iterativamente até ser encontrada uma

solução satisfatória. Por fim, a metodologia devolve as divisões identificadas com o respetivo módulo, bem como os centros dos *clusters* calculados, que são utilizados na configuração dos módulos.

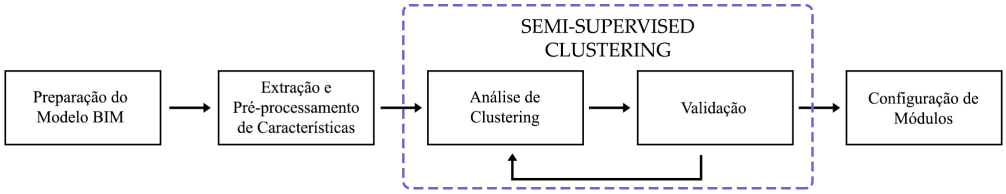

Figura 1
Metodologia proposta para a modularização de divisões.

A metodologia foi implementada sob a forma de *plugin* para o *software Autodesk Revit*, tendo sido utilizadas as linguagens de programação C# e *Python* durante o seu desenvolvimento.

3.1. Preparação do modelo BIM

Na base da metodologia apresentada, está o modelo BIM de um edifício. Antes de iniciar o processo de extração, este modelo requer uma prévia preparação por forma a cumprir as regras de modelação estipuladas. Nomeadamente, visto a modularização ser aplicada a divisões, a categoria *Room* do *Revit* deve estar devidamente modelada e identificada com o parâmetro *Uniclass Code* [11], por forma a distinguir diferentes tipologias de divisões no modelo. Embora a análise possa ser igualmente executada para diferentes tipos de divisões, não é recomendada a análise de múltiplos tipos simultaneamente, uma vez que as diferenças inerentes entre tipologias (e.g., casas de banho vs. cozinhas) podem facilmente ofuscar as diferenças entre divisões do mesmo tipo. Isto pode dar origem a resultados indesejados.

3.2. Extração e pré-processamento de características

Uma vez pronto o modelo BIM, a informação das divisões escolhidas é extraída para análise. A informação obtida inclui as características consideradas mais relevantes para descrever a arquitetura das divisões e pode ser classificada em três categorias: (1) *arquitetura da divisão*; (2) *dados sobre as tipologias de objetos*; e (3) *dados sobre o posicionamento dos objetos*.

A primeira categoria, *arquitetura da divisão*, considera o tamanho e forma do *Room*, incluindo medidas como o comprimento, a largura, a altura, a área, o perímetro e o seu número de paredes. Para tornar estas dimensões invariáveis à orientação da divisão, o comprimento e a largura são considerados, respetivamente, o valor mais alto e mais baixo entre os dois.

A segunda categoria, *dados sobre a tipologia de objetos*, descreve as características arquitetónicas da divisão (portas, janelas, mobiliário e equipamentos) através de informação quantitativa e descritiva dos objetos, tais como as suas tipologias no *software*

de modelação. A título de exemplo, consideremos as características das seguintes casas de banho:

- CB1 – uma sanita, um lavatório, um chuveiro;
- CB2 – uma sanita, dois lavatórios, um chuveiro, um bidé;
- CB3 – uma sanita, um lavatório, uma banheira.

Embora a casa de banho CB1 seja facilmente distinguível da CB2 pelo seu número de equipamentos sanitários, a mesma não é distinguível da CB3 apenas com esta informação. Para contabilizar esta diferença, foi desenvolvido um algoritmo de vectorização para codificar as diferentes tipologias de equipamentos sanitários em informação numérica, transformando os dados categóricos (tipos) em numéricos (vetores). Assim, é obtido um espaço multidimensional de características numéricas onde a análise de *clustering* pode ser aplicada. Retomando o exemplo anterior, as casas de banho seriam representadas pelos vetores visíveis na Tabela 1.

Tabela 1
Exemplo da conversão de variáveis categóricas em variáveis numéricas.

Instâncias de Casas de Banho		Chuveiro	Banheira	Sanita	Lavatório	Bidé	
Casa de banho CB1	[	1	0	1	1	0	]
Casa de banho CB2	[	1	0	1	2	1	]
Casa de banho CB3	[	0	1	1	1	0	]
...							
Casa de banho CBn	[	–	–	–	–	–	]

Finalmente, a terceira categoria, *dados sobre o posicionamento dos objetos*, retorna informação sobre o posicionamento dos objetos na divisão. Para isso, é calculada a distância euclidiana entre o ponto de posicionamento de cada objeto e o centro do *Room*, tornando esta informação independente da sua orientação. Usando este método, divisões simétricas ou transpostas têm maior probabilidade de serem incluídas no mesmo módulo, visto a distância ao centro permanecer igual. Várias métricas são depois derivadas das distâncias calculadas para caracterizar a divisão (i.e., média, desvio padrão, máximo, mínimo e somatório).

Com a informação extraída, os dados são normalizados entre 0 e 1 e enviados para análise.

3.3. *Semi-supervised clustering*

O *semi-supervised clustering* é um processo iterativo aqui aplicado em duas fases distintas: análise de *clustering*; e a validação. A primeira fase permite calcular a semelhança entre divisões, agrupando-as por *cluster,* enquanto a segunda permite ao utilizador examinar os *clusters* obtidos, validando as divisões conforme a sua correta/incorreta atribuição ao *cluster*.

3.3.1. Análise de *clustering*

A análise de *clustering* é baseada no algoritmo *Constrained K-means algorithm* (COP--KMEANS), introduzido por Wagstaff et al. [12]. O COP-KMEANS é uma variação do algoritmo K-means [13], com a incorporação adicional de restrições dadas pelo utilizador. Estas restrições traduzem o conhecimento do utilizador sobre as instâncias que devem ou não ser agrupadas. Este conhecimento é dado sob a forma de dois tipos de restrições: (1) *must-links*, que especificam duas instâncias que devem ser colocadas no mesmo *cluster*; e (2) *cannot-links*, que especificam duas instâncias que não devem ser colocadas no mesmo *cluster*.

Na iteração base, i.e., a primeira análise antes da validação, uma vez que ainda não existe *feedback* do utilizador (i.e., nenhuma restrição), o algoritmo COP-KMEANS regride para o algoritmo K-means. Apenas após o primeiro processo de validação (Secção 3.3.2), é que as restrições são criadas para guiar o COP-KMEANS.

Um requisito de ambos os algoritmos é a identificação do número de *clusters* k nos quais os dados devem ser agrupados – valor que nem sempre é conhecido. Para ultrapassar este problema, a implementação proposta avalia múltiplos números de clusters k_i, dentro de um dado intervalo $k_{min} \leq k_i \leq k_{max}$, onde $k_{min} \geq 2$ e $k_{max} < \overline{D}$. D representa os *Rooms* existentes no modelo BIM. Dentro deste intervalo, para cada conjunto de *clusters* é calculado o Coeficiente de Silhueta [14], sendo escolhido o k_i que retorne o maior coeficiente. Para um maior controlo, os valores de k_{min} e k_{max} podem ser alterados pelo utilizador.

3.3.2 Validação

O processo de validação permite ao utilizador ganhar controlo sobre os resultados do algoritmo, ajudando-o a identificar as diferenças relevantes entre módulos. De forma a evitar sobrecarregar o utilizador com informação, apenas uma pequena percentagem das divisões mais dessemelhantes (i.e., as divisões mais longe do centro do *cluster*) são mostradas durante o processo de validação. A percentagem de dessemelhança pode ser definida pelo utilizador.

Usando esta percentagem, as divisões consideradas mais dessemelhantes no *cluster* são validadas uma a uma, tal como ilustrado na Figura 2. Nesta interface, o utilizador pode:

1. Aceitar o *cluster* sugerido;
2. Rejeitar o *cluster* sugerido;
3. Atribuir a divisão a outro *cluster*, diferente do sugerido.

Figura 2
Interface de validação do plugin desnvolvido.

Aceitar ou atribuir uma divisão a um *cluster* específico (opção 1 e 3) cria uma restrição *must-link* entre a divisão validada e a divisão correspondente ao centro do *cluster* sugerido ou escolhido. Por sua vez, rejeitar o *cluster* sugerido (opção 2) cria uma restrição *cannot-link* entre a divisão validada e a divisão correspondente ao centro do *cluster* sugerido.

Com esta informação, a fase de validação termina, iniciando o processo iterativo entre o algoritmo COP-KMEANS, que sugere novos *clusters* tendo em conta o *feedback* do utilizador, e a validação, que avalia os novos resultados do algoritmo. Este processo pode ser repetido várias vezes até ser encontrada uma solução satisfatória. Esta solução pode ainda ser alterada numa revisão manual final, na qual o utilizador pode mover divisões entre *clusters*.

É importante referir que o *feedback* do utilizador é guardado entre iterações, ou seja, restrições criadas na primeira iteração permanecem até ao final do processo iterativo. Neste ponto, por forma a, mais uma vez, prevenir que o utilizador seja alvo de demasiada informação, um *Room* que já tenha sido previamente validado com um *must-link* não é mostrado em iterações seguintes, mesmo que se encontre dentro da percentagem de dessemelhança.

3.4. Configuração de módulos

Uma vez encontrada uma solução, é criado um módulo para cada *cluster*. Este módulo corresponde ao centro do *cluster*. Sendo que toda a informação sobre as divisões já foi extraída anteriormente, é também possível gerar automaticamente a geometria 3D do módulo de cada *cluster*, que pode ser usada para substituir as divisões não-modulares originais do modelo BIM.

4. Caso de Estudo

A metodologia proposta foi demonstrada num caso de estudo de um complexo residencial composto por seis blocos com um máximo de seis pisos. O projeto fez parte de um investimento em grande escala, incluindo até dois hectares de área construída, jardins privados e outras áreas de atividades exteriores.

Durante a fase de pré-construção, foi identificado um elevado potencial para a modularização e industrialização das 233 casas de banho do projeto, uma vez que estas apresentavam estruturas e sistemas de parede semelhantes. Adicionalmente, sendo esta uma das tipologias de divisões com mais instalações por metro quadrado, a sua modularização resultaria em vantagens significativas nos custos e duração do projeto. Esta modularização foi executada manualmente pelo empreiteiro de obra e consistiu em racionalizar o design do edifício para encontrar grupos de casas de banho que pudessem ser industrializadas na fabricação. Este processo manual é usado neste caso de estudo como referência para avaliar a metodologia proposta.

4.1. Modularização manual pela equipa de empreiteiros

A modularização manual consistiu em comparar a arquitetura de todas as 233 casas de banho do projeto de modo a encontrar semelhanças. Através desta comparação, as casas de banho foram agrupadas, primeiro, de acordo com a forma da divisão e respetivas dimensões e, segundo, de acordo com a disposição em planta dos equipamentos sanitários contidos. Dentro de um grupo ou *cluster* de casas de banho, foram admitidas pequenas diferenças, tais como casas de banho simétricas ou pequenas variações no espaçamento entre equipamentos.

No final deste processo, as 233 casas de banho únicas foram reduzidas a 46 grupos que podiam, com pequenas alterações ao projeto, ser adaptadas para industrializar a sua construção. Os resultados foram listados numa folha de *Excel*, com as respetivas identificações, dimensões e *cluster* atribuído a cada casa de banho. Esta informação foi também guardada num ficheiro CAD, contendo as plantas de todas as casas de banho agrupadas por *cluster,* bem como o respetivo módulo proposto para representar cada grupo.

Este processo manual foi complexo e demorado, requerendo uma equipa de duas pessoas e dois meses para ser finalizado. Permitiu, no entanto, reduzir significativamente os custos de construção e melhorar a produtividade com fabricação *offsite*.

4.2. Modularização automática usando a metodologia proposta

Tal com mencionado na Secção 3.1, todas as casas de banho foram identificadas com o seu parâmetro *Uniclass Code*, i.e., *SL_35_80_08*, permitindo a filtragem das divisões para a análise exclusiva de casas de banho. Usando estes dados, a informação das casas de banho foi extraída e processada automaticamente utilizando os métodos descritos anteriormente.

As análises de *clustering* foram executadas para o intervalo de 36-56 *clusters* (±20% que o número de *clusters* encontrado durante a modularização manual) e uma percentagem de dessemelhança de 60% (i.e., apenas as divisões com uma percentagem de dessemelhança superior a 60% são mostradas durante o processo de validação). Finalmente, foi iterativamente executado o processo de *semi-supervised clustering* até ser encontrada uma solução satisfatória.

Entre cada iteração da análise de *clustering*, foram calculadas várias métricas para medir o desempenho do algoritmo face à modularização manual, i.e., o *ground truth*. Entre elas, o *Normalized Mutual Information Score* (NMI) e o *Adjusted Mutual Information Score* (AMI) [15] medem a concordância entre duas atribuições de *clusters* (rótulos previstos vs *ground truth*), ignorando permutações. A diferença entre as duas é que o AMI é ajustado para considerar a aleatoriedade do algoritmo. Foram também usadas as métricas *Homogeneity* e *Completeness* [16], que permitem uma análise mais intuitiva dos resultados, i.e., a primeira métrica indica se cada *cluster* é apenas composto por membros da mesma classe, enquanto a segunda indica se todos os membros de uma dada classe estão classificados no mesmo *cluster*.

5. Resultados e Discussão

A Figura 3 resume os resultados da aplicação da metodologia ao caso de estudo, incluindo o valor de NMI, AMI, *Homogeneity* e *Completeness* por iteração, bem como o número de *clusters* obtido pelo Coeficiente de Silhueta. A Figura 4 mostra o número de validações por iteração.

Analisando a Figura 3, podemos observar que a iteração base (i.e., antes do processo de validação) atinge um NMI de 92,2% e um AMI de 85,6%. Estes resultados asseguram a viabilidade das características extraídas e na sua capacidade de retratar as casas de banho.

Após a primeira iteração, as sucessivas iterações mostram uma melhoria dos valores NMI e AMI em 4,2% e 7,7%, respetivamente, atingindo os valores máximos de 96,4% e 93,3% na iteração 3. Tanto a *Homogeneity* como a *Completeness* refletem estes resultados, apesar da pequena descida na iteração 2, que se deve ao rearranjo dos *clusters* em resposta às restrições criadas pelo utilizador durante a fase de validação. Focando a Figura 4, é possível ver que o número de módulos validados oscila devido aos incrementos de 5% na percentagem de dissemelhança, mas decresce gradualmente ao longo do processo de validação, refletindo a convergência do algoritmo.

Neste caso de estudo, o processo de *semi-supervised clustering* foi terminado pelo utilizador após a iteração 8, aceitando os resultados dessa iteração como satisfatórios. Os resultados deste processo podem ser gerados como os módulos representativos de cada *cluster* (utilizando o respetivo centro), alguns dos quais são visíveis na Figura 5.

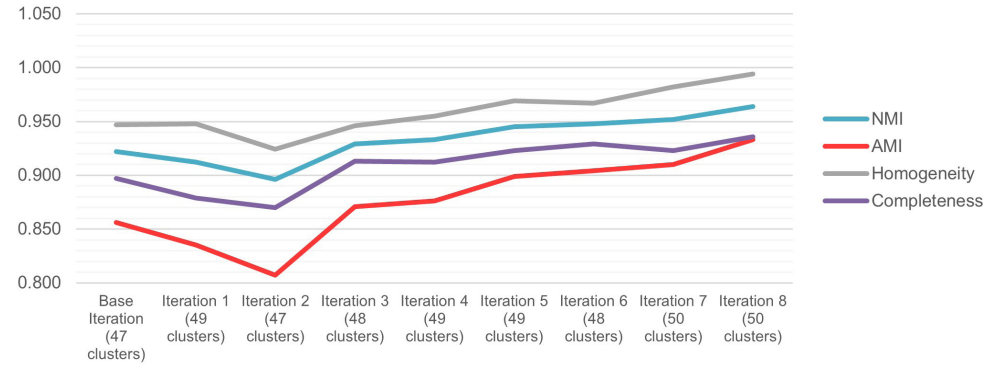

Figura 3
Performance da metodologia proposta comparada com a *ground truth*.

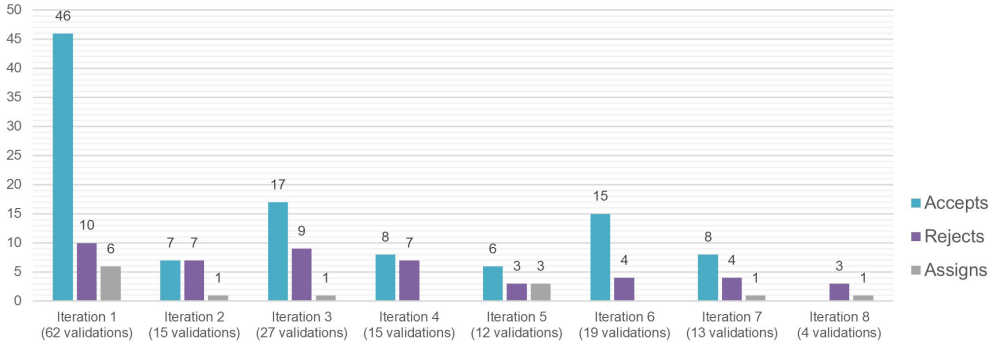

Figura 4
Número e tipo de validações por iteração.

Por fim, os ajustes finais aos módulos e aos respetivos *clusters* podem ser feitos manualmente. Dois modos de visualização foram criados para apoiar este processo:

O primeiro copia e gera as componentes base das divisões (e.g., paredes, portas, janelas, equipamentos, mobiliário) para uma nova vista no *software* de modelação, dispondo-as em grelha conforme o respetivo *cluster*. Isto permite uma comparação direta entre diferentes divisões e *clusters*, ajudando a identificar possíveis atribuições incorretas.

O segundo colora as divisões em planta de acordo com o respetivo *cluster*. Isto permite uma visualização mais contextualizada da envolvente de cada divisão, permitindo auferir informação que possa impactar significativamente a atribuição dos *clusters* como a localização de elementos estruturais próximos, possíveis falhas de continuidade em sistemas MEP, a arquitetura geral dos edifícios, entre outros.

No total, excluindo o processo de validação manual final, a aplicação da metodologia teve uma duração de duas horas. Comparativamente, como mencionado anteriormente, o processo de modularização manual teve uma duração superior a dois meses. Assim, os resultados obtidos representam uma redução de 99,6% da duração do processo de modularização, quando comparado à abordagem manual.

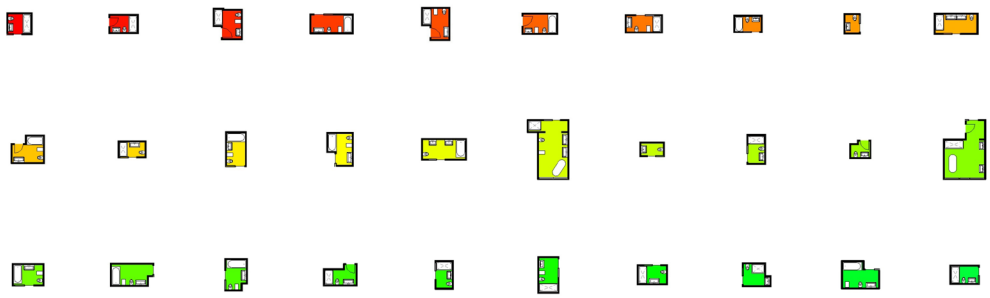

Figura 5
Exemplo de módulos obtidos com a aplicação da metodologia proposta.

6. Conclusões

O presente artigo teve como objetivo a melhoria do processo de modularização de projetos não modulares. Com este intuito, foi proposta uma metodologia baseada em BIM e *machine learning*, que permite automatizar o processo de modularização.

A aplicação da metodologia num caso de estudo demonstrou uma redução de 99,6% na duração do processo, mantendo um desempenho de 96,4% NMI e 93,3% AMI. Assim, a metodologia permite uma rápida avaliação do potencial de modularização de um projeto, bem como a aquisição de uma solução de modularização otimizada, tendo por base o *feedback* do utilizador. Desta forma, é possível condensar em escassas horas um processo que previamente demoraria semanas, justificando o desenvolvimento continuado desta metodologia.

Os trabalhos futuros irão focar-se na melhoria da precisão da metodologia, através do ajuste das características da análise de *clustering*, do teste da metodologia em diferentes tipologias de divisões e do desenvolvimento de ferramentas de apoio à integração da metodologia no fluxo tradicional da Construção.

Apesar de o artigo ter focado a modularização de casas de banho, procurou-se demonstrar que esta pode ser generalizada a outras divisões. Os autores esperam que o sucesso deste caso de estudo estimule o investimento na automatização das tecnologias no sector da Construção, fomentando a sua pesquisa e adoção.

7. Agradecimentos

Este trabalho é cofinanciado pelo Fundo Social Europeu (FSE), através do Programa Operacional Regional do Norte (Norte 2020) e do Programa Operacional Regional de Lisboa (Lisboa 2020) [Referência de Financiamento: NORTE-06-3559-FSE-000176 e LISBOA-05-3559-FSE-000014]. Os autores gostariam ainda de agradecer ao grupo CASAIS – Engenharia e Construção pelo seu contributo na realização deste artigo.

Referências

[1] Barbosa, F., Woetzel, J., and Mischke, J., *Reinventing Construction: A Route of Higher Productivity*. McKinsey Global Institute, 2017.

[2] Smith, R. E., *Prefab architecture: A guide to modular design and construction.* John Wiley & Sons, 2010.

[3] Song, L., Mohamed, Y., and AbouRizk, S. M., "Early contractor involvement in design and its impact on construction schedule performance," *Journal of Management in Engineering*, 2009, 25, pp. 12-20.

[4] Piroozfar, P., Altan, H., and Popovic-Larsen, O., "Design for sustainability: A comparative study of a customized modern method of construction versus conventional methods of construction," *Architectural Engineering and Design Management*, 2012, 8, pp. 55-75.

[5] B Bertram, N., Fuchs, S., Mischke, J., Palter, R., Strube, G., and Woetzel, J., *Modular construction: From projects to products*. McKinsey & Company: Capital Projects & Infrastructure, 2019.

[6] Construction, M. H., *Prefabrication and modularization: Increasing productivity in the construction industry*. SmartMarket Reports, 2011.

[7] Samarasinghe, T., Gunawardena, T., Mendis, P., Sofi, M., and Aye, L., "Dependency Structure Matrix and Hierarchical Clustering based algorithm for optimum module identification in MEP systems," *Automation in Construction*, 2019, 104, pp. 153-178.

[8] Isaac, S., Bock, T., and Stoliar, Y., "A methodology for the optimal modularization of building design," *Automation in Construction*, 2016, 65, pp. 116-124.

[9] Mohamad, A.; Hickethier, G.; Hovestadt, V.; and Gehbauer, F., "Use of modularization in design as a strategy to reduce component variety in one-off projects," em *21th Annual Conference of the International Group, Fortaleza*, Brazil, 2013.

[10] Tserng, H. P., Yin, Y. L., Jaselskis, E. J., Hung, W. C., and Lin, Y. C., "Modularization and assembly algorithm for efficient MEP construction," *Automation in Construction*, 2011, 20, pp. 837-863.

[11] NBS Enterprises, 'Uniclass 2015', 2021. [Online]. Disponível: https://www.thenbs.com/our-tools/uniclass-2015 [Acedido 20/10/2021].

[12] Wagstaff, K., Cardie, C., Rogers, S., and Schrödl, S., "Constrained K-means Clustering with Background Knowledge," em *Proceedings of 18th International Conference on Machine Learning*, United States, 2001.

[13] MacQueen, J., "Some methods for classification and analysis of multivariate observations," em *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, United States, 1967.

[14] Rousseeuw, P. J., “Silhouettes: a graphical aid to the interpretation and validation of cluster analysis,” *Journal of Computational and Applied Mathematics,* 1987, 20, pp. 53-65.

[15] Vinh, N. X., and Bailey, J. E. J., “Information theoretic measures for Clusterings comparison: Variants, properties, normalization and correction for chance,” *The Journal of Machine Learning Research*, 2010, 11, pp. 2837-2854.

[16] Rosenberg, A., and Hirschberg, J., “V-measure: A conditional entropy-based external cluster evaluation measure,” em *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, Czech Republic, 2007.