

ENSINO-APRENDIZAGEM DE SOFTWARE E REPRESENTAÇÃO DIGITAL EM CURSOS SUPERIORES DE DESIGN: A VISÃO DOCENTE

TEACHING AND LEARNING SOFTWARE AND DIGITAL REPRESENTATION IN DESIGN COURSES: THE TEACHERS' POINT OF VIEW

Marco Mazzarotto¹

Resumo

O objetivo deste trabalho foi coletar e analisar opiniões de professores de cursos superiores de design sobre os processos de ensino-aprendizagem de software e representação digital. Para isso, utilizou-se como método de pesquisa uma abordagem qualitativa implementada pela coleta de dados por meio de entrevistas presenciais e questionários online. Como resultado, foram encontradas opiniões sobre temas como a importância dos conhecimentos em software e representação digital, papéis e responsabilidades pelo processo de ensino-aprendizagem, modelos pedagógicos adotados, atividades didático-pedagógicas utilizadas, processos de avaliação, recomendações de mudanças necessárias e opiniões sobre o ensino-aprendizagem precisar ser mais integrado, contextualizado e interdisciplinar. Apesar dessa vontade identificada de tornar o ensino mais integrado e contextualizado com outros conhecimentos e práticas de design, principalmente às disciplinas e práticas de projeto, não foi encontrado nenhum modelo ou estratégia que permita de fato superar o isolamento e fragmentação do ensino desse conteúdo, que, portanto, permanece existindo em cursos de design.

Palavras-chave: ensino de design; representação digital; software gráfico; interdisciplinaridade; ensino contextualizado.

Abstract

The aim of this work was to collect and analyze the opinions of lectures in design courses on the teaching and learning processes of software and digital representation. For this, a qualitative approach was implemented by collecting data through face-to-face interviews and online questionnaires. The results found were opinions on topics such as the importance of knowledge in software and digital representation, roles and responsibility for the teaching-learning process, pedagogical models adopted, didactic-pedagogical activities used, evaluation processes, recommendations for necessary changes and opinions about teaching-learning need to be more integrated, contextualized and interdisciplinary. Despite this identified desire to make teaching more integrated and contextualized with other design knowledge and practices, mainly design project, no model or strategy was found to actually overcome the isolation and fragmentation of teaching of this content, which therefore remains in design courses.

Keywords: design teaching; digital representation; graphic software; interdisciplinarity; contextualized teaching.

¹ Professor Doutor, UTFPR - Departamento de Desenho Industrial – Curitiba, PR, Brasil.
marcomazzarotto@gmail.com; ORCID: 0000-0003-4473-0251

1. Introdução

O presente artigo teve como objetivo coletar e analisar opiniões de docentes sobre o processo de ensino-aprendizagem de representação digital em cursos superiores de design. Essa pesquisa faz parte de um projeto de investigação maior, que busca entender as práticas atuais de ensino-aprendizagem, principalmente identificando seus problemas de isolamento e pouca integração e contextualização com outras práticas e conhecimentos em design. Entendimento este que deve servir de ponto de partida para buscar formas de superar o isolamento e promover no futuro uma maior integração, contextualização e interdisciplinaridade no ensino.

No trabalho aqui apresentando, o foco esteve em levantar de forma ampla opiniões dos docentes sobre o processo de ensino-aprendizagem de software e representação digital, para depois entrar na questão específica sobre a integração e contextualização desse ensino com outras práticas. A pesquisa teve uma abordagem qualitativa e se deu por meio de entrevistas presenciais e questionários online com dezesseis docentes de seis diferentes instituições. Como resultado, foram encontradas opiniões convergentes e divergentes sobre temas como a importância dos conhecimentos em software e representação digital, responsabilidade pelo processo de ensino-aprendizagem, modelos pedagógicos adotados, atividades didático-pedagógicas utilizadas, processos de avaliação, mudanças necessárias e, como não poderia faltar nesta pesquisa, as opiniões sobre o ensino-aprendizagem poder ser mais integrado e contextualizado. Nesse último ponto identificou-se que, apesar de existir uma concordância dos professores sobre a importância de tornar o ensino menos isolado, ainda não foram encontrados modelos ou estratégias que de fato permitam promover essa integração com outros conhecimentos.

Estruturalmente, este artigo está dividido em mais quatro partes: (1) uma revisão de literatura que busca explicar melhor a importância da representação digital para o design, como esse conhecimento tende a ser abordado de forma pouco contextualizada nos cursos de design e os problemas que esse isolamento pode causar; (2) a descrição detalhada dos procedimentos metodológicos adotados; (3) a apresentação e discussão dos resultados coletados, dividida nos sete temas identificados; (4) e as considerações finais e desdobramentos futuros.

2. Design, Representação Digital e o Ensino-Aprendizagem Fragmentado

Para Goldschmidt (2004), a representação é uma ferramenta imprescindível para a prática do design, já que as ideias precisam ser representadas para que possam ser compartilhadas, observadas e compreendidas pelos outros, mesmo que o outro seja o próprio designer.

Diferentes tipos de representação podem, segundo Matté (2009), ser encontradas ao longo do processo de design. Nas etapas iniciais, a rapidez e a liberdade do desenho manual fazem essa técnica ser muito adequada para esse momento exploratório e de pensamento divergente. Porém, conforme as decisões projetuais vão sendo definidas, as representações gráficas evoluem para uma linguagem mais detalhada e estruturada, apta para a produção, documentação e comunicação final. Nessas etapas, o uso do computador e de software gráfico são de grande importância, pois como afirma Rosselli (2012), essas tecnologias permitem grande precisão e exatidão na execução das tarefas, possibilidade em corrigir e reproduzir inúmeras vezes os desenhos, facilidade de armazenamento e distribuição dos documentos e uma maior facilidade na execução e simulação de técnicas que antes demandavam grande habilidade manual.

Ou seja, a representação digital, desenvolvida através do uso de software gráfico, se mostra como de grande importância para a prática do design. Importância essa que é supervalorizada pelo mercado, como apontam pesquisas de Marshall & Meachem (2007), Yang, You & Chen (2005), Lynn (2006), Aldoy (2011) e Atharifar, Yildiz & Knapp (2013). Porém, essa supervalorização não costuma encontrar sempre respaldo no corpo docente dos cursos superiores de design, pois como afirma Lynn (2006), há professores de design que acreditam que o uso do computador diminui a qualidade das soluções de design. Opinião semelhante é encontrada em Lauer (2015), que afirma que há visões pedagógicas dentro dos cursos de design que defendem minimizar o uso do computador durante o processo criativo, para que os estudantes possam trabalhar primeiro só com a ‘ferramenta da mente’.

Nesse contexto, o cenário que se desenha é de um lado o mercado de trabalho e estudantes de design depositando grandes demandas nos cursos de graduação para fornecerem formação adequada no uso de software gráfico. Do outro lado, as instituições de ensino e seu corpo docente ou não compartilhando desse mesmo entusiasmo ou não tendo diretrizes claras sobre como integrar esse conhecimento com o restante da formação em design. A consequência disso, segundo Marshall & Meachem (2007), é que a inserção do ensino de software em cursos de design no final da década de 90 foi mais reativa do que proativa. Para as autoras, as instituições inseriram esses conteúdos em seus currículos como uma resposta às pressões de alunos e do mercado, e não devido a um entendimento de como seria a melhor forma de integrá-los dentro da formação em design. Como resultado, o ensino desses conteúdos foi na maioria das vezes estruturado na forma de disciplinas isoladas e completamente separado da prática projetual que ocorre nos estúdios, ateliers e outros espaços da universidade. No contexto brasileiro, Spitz (1995) identificou o mesmo problema, apontando que embora a computação gráfica tenha se consolidado como uma disciplina obrigatória nos currículos brasileiros, é comum encontrá-la totalmente desvinculada das demais. Essa fragmentação e descontextualização no ensino de representação digital foi identificada e confirmada na análise dos currículos realizada por Mazzarotto & Ulbricht (2016).

A principal crítica a esse ensino fragmentado é que ele pode até formar um aluno com grandes habilidades na operação do software, mas isso não é suficiente para formar designers que saibam como aplicar a representação digital adequadamente em projetos de design. Saber apenas utilizar o software para criar imagens fora de contexto não é suficiente para também saber transferir esses conhecimentos na resolução de problemas de design. Como afirma Marshall (2004), conhecimentos instrumentais no uso do software, desconectados de preocupações com habilidades de resolução de problemas e com a sua relação com os demais aspectos da comunicação visual, é, utilizando uma metáfora, o equivalente a uma pessoa que, mesmo conhecendo todas as palavras, é incapaz de formar frases conexas e se fazer entender. Ou como coloca Stout *apud* Marshall & Meachem (2007), é fundamental integrar as competências já tradicionais do design com as novas tecnologias, caso contrário corre-se o risco de criar – usando metáforas e uma linguagem informal – ‘macacos de computador’, muito bons em usar as tecnologias, mas deixando o ‘rabo balançar o cachorro’, já que ao não terem um entendimento claro de como organizar textos e elementos para criar uma mensagem eficaz, deixam o computador cuidar disso por eles.

Por essas razões, justifica-se a relevância em buscar formas de superar essa fragmentação no ensino de software e representação digital. Antes, porém, esta pesquisa buscou entender melhor os processos atuais de ensino-aprendizagem dessa temática a partir da perspectiva docente. Como essa pesquisa foi conduzida e estruturada é apresentado a seguir.

3. Procedimentos Metodológicos

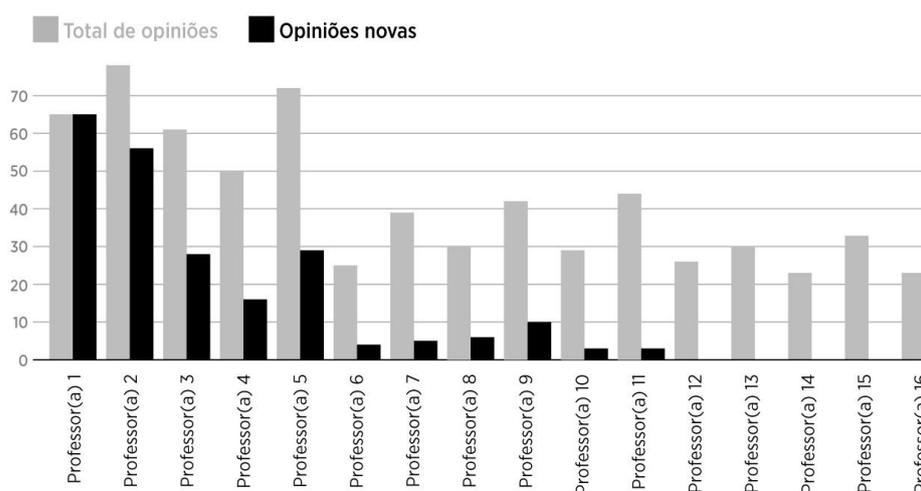
O presente trabalho foi estruturado a partir de uma abordagem qualitativa de pesquisa. Ou seja, o interesse maior não esteve em descrever todas as opiniões e conclusões encontradas de forma estatística e generalizável, porém superficial, como costuma ocorrer em abordagens quantitativas. Pelo contrário, o objetivo foi aprofundar o entendimento sobre as opiniões dos docentes sobre os temas abordados, mesmo sabendo que a quantificação dessas opiniões não pode ser generalizada para toda a população docente.

A coleta dos dados se deu por meio de entrevistas presenciais e questionários on-line com docentes de cursos de design, principalmente aqueles envolvidos com o ensino de software e representação digital ou de áreas nas quais esse conteúdo é demandado, como é o caso das disciplinas de projeto. Participaram ao todo dezesseis professores de seis diferentes universidades públicas (UTFPR, UFPR, UDESC, UFSC, UFRN e FAU/USP), com idades variando entre 28 e 55 anos e com experiência docente entre 1 e 30 anos. A seleção dos participantes foi do tipo não-probabilística por conveniência.

A definição do tamanho da amostra se deu por saturação. Como afirma Fontanella et al (2011), na amostragem por saturação a coleta de dados é interrompida quando se constata que os novos casos abordados não apresentam mais nenhuma contribuição para a teorização almejada. Para comprovar essa saturação, é importante que a cada nova entrevista os dados sejam analisados, condensados em temas e comparados com as análises anteriores. Se temas e opiniões novos não surgirem, é sinal de que a saturação da amostra foi alcançada.

Na Figura 1, é apresentada a análise e contagem das opiniões de cada participante (na cor cinza) e a contagem de quantas dessas opiniões eram novas e diferentes de opiniões já encontradas anteriormente (na cor preta). Como pode ser observado, novas opiniões pararam de ser encontradas a partir do décimo primeiro participante, justificando a possibilidade de encerrar a coleta de dados.

Figura 1: Saturação das opiniões ao longo da amostra.



Fonte: Elaborado pelo autor.

As entrevistas foram conduzidas de forma semiestruturada em encontros presenciais com os cinco primeiros participantes. Para aumentar a diversidade da amostra, permitindo

levantar a opiniões de professores de outros estados brasileiros, os outros onze participantes responderam questionários online enviados por e-mail. Em ambas modalidades, os tópicos abordados foram similares: a importância do ensino de software e representação digital, de quem é a responsabilidade pelo ensino-aprendizagem desses conteúdos, as estratégias de ensino, as mudanças que acredita ser necessárias nos modelos atuais e a opinião sobre formas contextualizadas de ensino-aprendizagem dessa temática.

Por fim, a análise dos dados seguiu o processo de análise qualitativa proposto por Miles, Huberman & Saldaña (2014), consistindo na leitura dos dados brutos, a condensação de trechos longos em códigos menores, a aplicação destes códigos em trechos similares, o agrupamento destes códigos em categorias e temas maiores e a redação de memoriais descritivos articulando essas descobertas. Ou seja, utilizando a codificação para identificar padrões entre as falas que puderam ser agrupados e depois condensados em temas. Trechos das falas relevantes que ajudassem a entender a aprofundar os entendimentos sobre esses temas também foram mantidos e utilizados *ipsis litteris*. Referente ao processo de codificação, ele foi do tipo *bottom-up*, definido pelos autores como a estratégia de criar códigos conforme eles emergem dos dados analisados, sem categorias pré-definidas previamente.

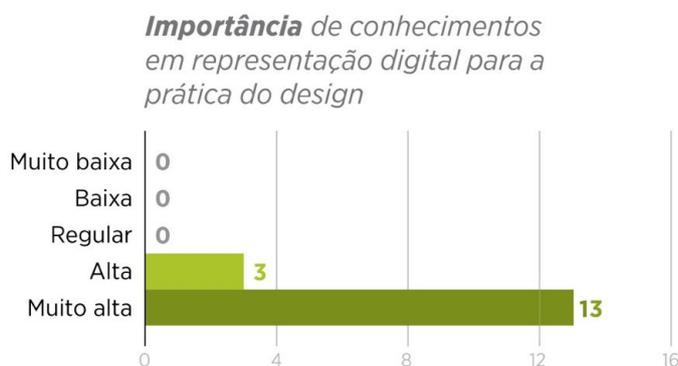
4. Resultados e Discussão

Neste tópico são apresentados e debatidos os dados coletados junto aos docentes. Para organizar melhor a apresentação dos resultados, cada tema será apresentado e analisado em um subtópico específico, totalizando sete temas distintos.

4.1. A Importância de Possuir Conhecimentos em Software e Representação Digital

A primeira questão abordada com os participantes era referente a quão importante é ter conhecimentos em software e representação digital para a prática do design. Nessa pergunta, inicialmente os respondentes deveriam escolher entre as opções de resposta: muito baixa, baixa, regular, alta e muito alta – para depois darem mais explicações sobre sua escolha. Como pode ser observado na Figura 2, treze professores consideraram essa importância muito alta e três a consideraram alta.

Figura 2: Opinião de professores sobre a Importância de conhecimentos em representação digital para a prática do design.



Fonte: Elaborado pelo autor.

As justificativas para essa importância giraram em torno de dois grupos de respostas: a demanda do mercado e as contribuições da ferramenta para a atuação profissional.

A **demanda do mercado** foi mencionada por doze dos dezesseis professores. Em linhas gerais, o que eles apontam é que existe pouco espaço no mercado de trabalho para quem não domina representação digital e o software a ela relacionados, principalmente para quem está começando. Sobre isso, uma professora afirmou que o processo de seleção de vagas de trabalho “...é uma peneira, e a primeira peneira é se a pessoa sabe ou não usar software”, na mesma linha, outro professor apontou que “no caso do software gráfico, acho que o mercado impõe pesadamente”. Mesmo para estudantes na busca por um estágio, uma professora afirmou que “... até mesmo nas entrevistas que a gente acompanha, estágio. Todos eles exigem já que os alunos tenham o domínio disso, então eles na entrevista já exigem”. Um professor também comentou uma supervalorização do software dependendo da área de atuação: “As vezes dependendo da área que você vai trabalhar, tipo agência de publicidade, parece que o software é o principal, claro tem a criatividade, você tem que resolver os problemas, mas como a pressão de tempo é muito grande, o tempo de execução é muito baixa, e daí o conhecimento de software é importante”.

Porém, essa grande valorização dos conhecimentos em representação digital pelo mercado também é abordada de forma crítica por parte dos professores, que apontam que esses conhecimentos não são necessariamente os mais importantes e não substituem outras habilidades. Como afirmou uma professora: “Pensando em design gráfico acho que todos vão precisar de software em algum momento. Mas não adianta nada a pessoa saber software e não saber projeto, é uma combinação dos dois. Acho que o software pode condicionar a maneira de pensar, e pode dar a falsa sensação de que a pessoa sabe fazer projeto”. De forma mais crítica, outra professora afirma que “software você ensina para qualquer um, você senta com uma pessoa três dias e ela começa a se defender e fazer, mas design não é assim”. Por fim, um professor inclusive afirmou que a importância dada ao software pode ser demasiada e não refletir o perfil ideal de um designer: “Pelas vagas de emprego que são divulgadas, o pessoal pede domínios e domínios de software que não necessariamente tem a ver com a capacidade produtiva do designer, não tem nada a ver com a ação de projetar, mas sim com a ação de executar. Que não necessariamente caberia ao design então eu acho que é ainda uma falta de compreensão do design no Brasil”.

Mas além da demanda do mercado – e essa controvérsia se existe uma supervalorização do software por este – também foram apontadas **contribuições importantes de conhecimentos em representação digital para a formação do designer**. Essas contribuições foram expressadas através de dois grupos principais de opiniões: o primeiro referente ao fato da representação digital ser uma entre as várias formas de representação disponíveis e necessárias para a prática atual do design, e que dificilmente pode ser dispensada no contexto atual da profissão; e o segundo referente a importância de conhecer e saber usar de forma adequada e eficiente o software, evitando – na palavra dos entrevistados – “enjambres” ou “gambiarrras” que podem atrapalhar ou atrasar o projeto.

Sobre o primeiro grupo de opiniões, o que se colocou é que atualmente a adoção generalizada do computador nos processos de design praticamente impõe ao designer também conhecer essa forma de representação, que é diferente e complementar às demais. Como afirmou uma professora, “quanto mais mídias você domina, mais você vai conseguir se mover, o que eu digo para o pessoal é que cada mídia tem possibilidades representativas que são diferentes entre si, quanto mais mídias você domina, mais você consegue se mover dentro de diversos campos representativos, fica menos limitado”. Mais do que uma opção, alguns

professores inclusive afirmaram que no cenário atual da prática do design é inviável não possuir esses conhecimentos: *“nenhum designer pode sair da universidade sem saber mexer”, “no caso do design gráfico acho bem difícil e não consigo ver gente hoje dispensando essa ferramenta”*.

Quanto ao segundo grupo de opiniões, referentes à importância em dominar software para representação digital de modo a tornar o trabalho projetual mais eficiente, percebe-se uma ligação destes argumentos com o conceito de **conhecimento estratégico**. Para Pinto (1998), estes são os conhecimentos que permitem definir quais outros conhecimentos são mais adequados para cada situação, tendo uma natureza estratégica que busca otimizar a realização das atividades. O que esses professores defendem é que não basta saber *“se virar”* no software, é preciso entender para qual objetivo cada software funciona melhor, quando utilizar cada um, quais os fundamentos por trás do seu uso e qual a forma mais eficiente de usar cada ferramenta. Como afirmou um professor, o tempo de execução é um parâmetro importante, *“porque você pode ter dois resultados exatamente iguais, um levou 15 minutos o outro levou 5 horas, qual tá certo? Não é só o parâmetro visual, tá certo o que levou 15 minutos, tá errado o cara que levou 5 horas pra fazer. Ele resolveu, mas dando voltas”*. Da mesma forma, é importante conhecer as opções de ferramentas e para que serve cada uma: *“eu acho que não pode ser assim, se vira, usa o software que você quiser. Eu acho que você cria um profissional deficitário, ele vai lá e vai fazer o enjambre que der”*, ou como afirma outra professora *“como que você vai fazer um livro de trezentas páginas no illustrator? [software vetorial mais indicado para publicações com poucas páginas]. Ainda nessa questão, também existe a importância em saber operar a ferramenta de forma correta e organizada, para que outros profissionais ou até mesmo o próprio designer consigam entender e trabalhar posteriormente neste arquivo: “se você não cobra que eles façam organizado, talvez nem eles mesmo entendam a estrutura” ou “estou ensinando pra eles desenvolverem de forma correta para no caso de outra pessoa conseguir editar ela consiga se achar no arquivo”*.

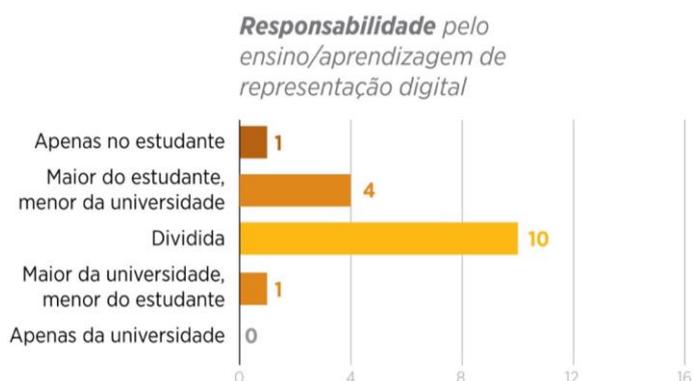
4.2. Responsabilidade pelo Processo De Ensino-Aprendizagem

Após essa importância da representação digital para a formação em design ser apontada e debatida, a segunda parte da entrevista migrava para a discussão sobre **qual seria o papel da universidade e o papel do estudante** no processo de ensino/aprendizagem destes conhecimentos. O objetivo foi identificar quem o entrevistado achava ser o principal responsável pelo processo: se de um lado ele era exclusividade do estudante, que deveria encontrar sozinho formas de aprender estes conhecimentos; se, do outro lado, era de grande responsabilidade da universidade, que deveria estruturar o curso e as disciplinas de modo a abordar todos os conteúdos necessários sobre representação digital; ou uma divisão de responsabilidade entre os dois. Como pode ser visto na Figura 3, a maior parte dos entrevistados (dez no total), considerou que fazer acontecer a aprendizagem em representação digital é uma **responsabilidade igualmente dividida entre o estudante e sua instituição de ensino**. Já quatro consideraram que ela pende mais para o estudante, um professor considerou que ela é apenas do estudante e outro que ela pende mais para o lado da instituição.

Independente da resposta escolhida, porém, as justificativas dadas pelos professores foram em linhas gerais muito próximas. Todos apontaram que a universidade tem um papel que não pode desconsiderar para que a aprendizagem de representação digital ocorra de forma adequada, até mesmo a professora que no primeiro momento frisou que a responsabilidade é apenas do estudante, defendeu – de forma até contraditória – ações que a

instituição deve adotar. Da mesma forma, todos apontam para a incapacidade das disciplinas abordarem todo o conteúdo necessário e da necessidade de formar estudantes proativos e que saibam também procurar complementar e aprofundar seu aprendizado. O que muda nas respostas é o grau de iniciativa que se espera que os estudantes tenham e a quantidade de aulas, conteúdos e atenção que o curso deve oferecer.

Figura 3: Percepção dos docentes sobre a responsabilidade pelo processo de ensino/aprendizagem de representação digital.



Fonte: Elaborado pelo autor.

Quanto ao **papel da universidade**, as respostas defendem que o curso precisa apresentar o básico, mostrar onde começar os estudos, indicar caminhos, dar uma visão geral de todos os tipos de software e para que serve cada um, instigar a curiosidade e mostrar possibilidades, apresentar as tecnologias mais atuais do mercado, dispor professores para orientar e tirar dúvidas, ser uma ponte e um local de troca, apresentar problemas projetuais que demandem o uso desses conhecimentos, cobrar e avaliar o aprendizado e até mesmo oferecer infraestrutura para aqueles estudantes que não tem acesso ao computador fora da instituição. Nenhum professor defendeu que cabe a universidade fornecer todo o conhecimento necessário nessa área.

Quanto ao **papel do estudante**, cabe a ele, a partir do contexto básico ofertado pela instituição, buscar se aperfeiçoar, exercer sua autonomia, trazer suas questões e buscar orientação, ou, na palavra de alguns entrevistados: *“correr atrás”* da sua aprendizagem.

O que muda entre as respostas, é o grau de responsabilidade atribuído a cada sujeito deste processo. Para alguns professores, deixar o estudante com muita liberdade e autonomia pode ser um problema: *“Sozinho ele vai aprender a fazer enjambre”, “Isso pode ser ruim, você não sabe por onde começar, você não tem nenhum norte pra onde ir”, “se ficar só no estudante, pode ser que ele se bata mais, que ele demore um ano pra descobrir uma coisa que talvez se tivesse a figura do orientador, ele só falasse ‘oh que tal você ver isso’” ou “o nosso estudante ainda não tem o perfil de ser autossuficiente”*.

De maneira aposta, outro grupo de professores defendeu fortemente que cabe ao estudante *“correr atrás”*, sendo o papel da instituição apenas dar um ponto de partida e avaliar os resultados: *“[a responsabilidade] é do estudante, é dele, ele tem que correr atrás. É 80% dele, eu entendo que tem que ter em sala, a universidade tem que dispor daquele software, mostrar que lá você consegue fazer, mas é obrigação deles” ou “o que eu estou defendendo é que a universidade tem que ter um envolvimento com isso, mas é mais na*

avaliação, na cobrança. [...] Se a pergunta é a universidade tem que dar os instrumentos? Eu acho que não, mas se a pergunta é a universidade tem que se envolver nisso? Eu acho que sim [...] a universidade tem que estar muito envolvida, mas não ensinando os comandos dos programas, ela tem que estar cobrando”.

Há também o “**caminho do meio**”, como apontado por uma professora: “Acho que sempre é o caminho do meio, aí é filosofia de vida, o caminho do meio é o equilíbrio entre as coisas”. Nesse grupo há respostas que tentam equilibrar melhor as responsabilidades, sem jogar tanto para um dos sujeitos do processo, seja o estudante ou os professores/instituição: “Acredito na universidade como ponte e local de troca, o estudante precisa estar disposto a se envolver e o professor a ser um facilitador” ou “acho que a universidade tem que pincelar um pouco de tudo e mostrar possibilidades para eles, aí ele vai atrás do resto. Tem que indicar caminhos e ensinar ele a ter autonomia, dar um repertório amplo para ele, mas não dar ênfases”.

De modo geral, o que se percebe de comum entre os professores é que a universidade precisa apresentar um ponto de partida, caminhos, possibilidades e fornecer suporte e orientação ao longo do processo, e cabe ao estudante desenvolver autonomia para trilhar esses caminhos e desenvolver sua aprendizagem. Mesmo com a maioria dos professores escolhendo a opção dividida entre as alternativas apresentadas para esta pergunta, há uma clara falta de consenso sobre como é essa divisão, e quão presente e estruturado deve ser esse suporte por parte da universidade e o quanto de autonomia se deve exigir do estudante.

4.3. Dois Modelos de Ensino: Tutorado Versus Tentativa e Erro

Após essa discussão sobre o papel da universidade no processo de ensino aprendizagem, o tema seguinte a ser debatido passava a focar na **prática pedagógica** e nos modelos de implementação do ensino pela instituição. Nas falas, notou-se dois caminhos possíveis e opostos, mas o que não significa que não possam ser usados em conjunto ou em momentos diferentes da mesma disciplina. O primeiro caminho é uma aula com conteúdos, ferramentas e atividades mais estruturadas, controladas e centralizadas no professor, que denominaremos de modelo **tutorado**, enquanto o segundo caminho é uma aula mais livre, mais centralizada em um papel autônomo do estudante em procurar os conhecimentos necessários e descobrir como as ferramentas funcionam, modelo o qual chamaremos de **tentativa e erro**.

No **modelo tutorado**, as aulas são baseadas na resolução de exercícios práticos envolvendo o uso do software, é comum o professor realizar uma aula expositiva sobre as ferramentas envolvidas e uma demonstração inicial de como elas funcionam e/ou de como o exercício pode ser resolvido, também é comum existir um tutorial (em vídeo demonstrativo ou em um texto passo a passo) para o uso das ferramentas e/ou resolução do exercício. Aos estudantes então é dado tempo na aula para a realização do exercício em sala ou mandado para ser realizado em casa, por fim, cabe ao professor acompanhar a realização, tirar dúvidas e avaliar os resultados.

Sobre esse modelo tutorado, uma professora relatou:

Eu começo sempre, todas minhas aulas faço assim, eu passo todo o conteúdo daquela aula, passo comandos ou ferramentas, mostro na tela onde que são as coisas, no caso do software Adobe tem três lugares diferentes para acessar a mesma ferramenta. Então eu mostro, cada aluno tem uma forma diferente de ver o software, aí depois eu dou uma atividade para o resto da aula, que são mais três aulas. Então na primeira eu passo a

teoria, e nas outras eu vou fazendo as atividades, aí eu vou indo um por um, porque eles não absorvem só porque eu falo, aí enquanto eles vão mexendo eles vão tendo as mesmas dúvidas, aí eu vou mostrando direto no computador. [...] [Como material didático] a única coisa que eu faço é que eles têm todas as minhas aulas, que eu mando para eles sempre, que é um PDF com o resumo de todas as ferramentas e o passo a passo para eles saberem se virar em casa.

Já no **modelo tentativa e erro**, as aulas são voltadas para a proposição de exercícios sem um passo a passo prévio de resolução, cabendo ao estudante explorar a ferramenta, procurar tutoriais e outras fontes de informação que o auxiliem, testar caminhos e alternativas, e acabar errando no meio do caminho até desenvolver formas próprias de resolver o problema proposto. O papel do professor é apresentar uma explicação básica sobre o software e suas principais ferramentas, criar propostas de atividades para serem solucionadas, incentivar os estudantes a procurarem as soluções por si próprios e auxiliar aqueles que os procurarem com dúvidas.

Sobre esse modelo de tentativa e erro, um professor relatou:

Hoje, eu não dou aulas instrumentais de computação gráfica, eu não passo comandos, de CAD que deve ter uns 256 comandos eu passo cinco ou seis, que são assim bem fundamentais. [No primeiro dia da disciplina tem] uma aula expositiva. Ultimamente não tenho nem usado projeção, escrevo no quadro, o filet mignon, a essência de como é que faz [...] [Depois passo] propostas de execução de exercícios. Você tem desenhos para reproduzir. [...] Eles têm que procurar, eu não sugiro nada. Aí eles procuram da forma que quiserem, [...] o que eu faço, eu desmistifico a interface do software, eu faço as propostas de resolução, e tento observar de que maneira eles estão tentando resolver os problemas. É claro que é sempre um risco porque tem os que simplesmente não querem resolver, mas aqueles que se interessam e pesquisam trazem o feedback de que olha é possível, a partir das suas informações iniciais, ir muito longe [...] O que está embutido nessa forma de ensinar é a ampla variedade de tutoriais e materiais que têm na internet, é estrondosa, é da ordem de milhões. [...]o aluno aprende a se virar, não tem uma tutela ali”

É possível notar ligações diretas entre estes dois modelos com as duas posições sobre os papéis da universidade e dos estudantes debatidas anteriormente. Professores que defendem uma ação mais presente e estruturada da universidade tendem a adotar o modelo de aula tutorada, mais próximo da instrução direta, enquanto os professores que defendem que cabe ao estudante “*correr atrás*” do seu aprendizado são mais adeptos do modelo tentativa e erro, mais relacionado com as metodologias ativas.

Os que defendem o modelo tutorado, apontam que ele é mais efetivo e mais viável de ser realizado pelos estudantes: “Acho que dar muita liberdade nos exercícios, acaba deixando muito aberto e eles não se esforçam para fazer. Em geral isso é bem visível, os alunos que estão mais dedicados a fazer os exercícios em geral os resultados nos projetos são mais claros, você percebe que os arquivos vêm mais organizados, mesmo a qualidade do uso das técnicas. Fazer uma máscara de recorte bem-feita, isso faz bastante diferença”.

Já as críticas a esse modelo giram em torno das aulas muito tutoradas correrem o risco de ficarem monótonas e sem engajamento dos estudantes: “Eu abro o software e vou mostrando passo a passo e eles vão me seguindo passo a passo. Então é uma aula bem maçante, eles não aguentam muito tempo, eu não gosto muito, eu particularmente não gosto

e acho que o rendimento vai caindo com o tempo” ou “essa parte [expositiva] do início, muita pouca gente presta atenção, mas eu não sei como começar sem fazer isso. [...] Tem muito aluno que pede para eu ir fazendo com ele na tela, mas acho que vai dar a mesma coisa, alguns olham, outros não, alguns conseguem acompanhar”.

Já para o modelo de tentativa e erro, a defesa gira em torno da incapacidade da universidade em abordar todo o conhecimento sobre representação digital, e que, portanto, é mais importante focar em desenvolver nos estudantes senso crítico e a habilidade de “*correram atrás*” desses conhecimentos de forma autônoma: “*Não dá também pra universidade passar tudo, se ficar focado só nisso perde a parte importante de formar o profissional que pense aquilo que ele está fazendo*” ou “*Mais importante que ensinar ferramenta, é mostrar as várias possibilidades, dar introduções, o foco é sensibilizar o olhar e ensinar espírito crítico, ensinar o aluno a pensar e ter autonomia, do que ensinar ferramenta e dizer faz assim ou assado. Também é importante ensinar, despertar o aluno a estabelecer relações. Eu percebo que eles têm dificuldade em fazer relação com as outras disciplinas. Em um processo de ensino tem coisas mais importantes, a ferramenta é importante, mas acho que o foco não é a ferramenta*”.

Quanto às críticas a esse modelo, alguns respondentes apontam a dificuldade dos estudantes em trabalharem com tanta liberdade e de serem protagonistas no processo: “*Antes [...] eu deixava eles livres, não tinha uma roteirização disso. [...] O que eu percebi, veja só, o antes e o depois, eu percebia que eles ficavam muito aleatórios sem nenhum tutorial. Eu só dizia o que deveria ser feito, as ferramentas que deveriam, que poderiam ser utilizadas nesses exercícios. Aí eu via que o pessoal ficava assim, meio que mexendo no facebook, muito aleatório*”.

Encerrando esse tema, o que se percebe é a existência de dois grupos de opiniões, o primeiro muito baseado na instrução direta, que defende um modelo tutorado para o ensino-aprendizagem de representação digital e que vê uma responsabilidade maior da universidade nesse processo. Já o segundo, de forma oposta, é mais baseado em metodologias ativas, defende um modelo de tentativa e erro e vê uma maior responsabilidade do estudante para com a sua aprendizagem.

4.4. Tipos de Atividades Didático-Pedagógicas

A Tabela 1 apresenta o compilado de atividades pedagógicas que os professores utilizam em suas aulas para o ensino-aprendizagem de software e representação gráfica. Como pode ser visto, o uso de **tutoriais** foi a estratégia mais citada. Levando em consideração tanto tutoriais em vídeo quanto passo a passo, catorze dos dezesseis entrevistados fizeram menção a esse material didático, seja porque os utilizam diretamente em suas aulas, seja porque incentivam a sua procura e uso. Quanto à produção destes materiais, quatro professores produzem seus próprios tutoriais; um afirma que gostaria de produzi-los, mas não tem tempo; enquanto sete utilizam tutoriais encontrados na internet ou incentivam que os estudantes procurem por eles.

Tabela 1: Atividades pedagógicas utilizadas pelos professores

Atividade pedagógica	Quantidade de professores que utilizam
Tutoriais	14

Atividade pedagógica	Quantidade de professores que utilizam
Demonstração pelo professor	11
Projetos de design	11
Orientação individual	11
Exercícios práticos	9
Educação por pares	8
Tentativa e erro	6
Prova teórica	1
Demonstração por alunos	1

Fonte: Elaborado pelo autor.

Esse incentivo pela busca dos tutoriais, assim como a não obrigação destes professores em precisarem fazer novos, só é possível, segundo os respondentes, pela grande quantidade desses materiais disponibilizados gratuitamente na internet: *“O que está embutido nessa forma de ensinar: a ampla variedade de tutoriais e materiais que [...] tem na internet, é estrondosa, é da ordem de milhões”*.

O tutorial é uma estratégia de ensino importante tanto para o modelo tutorado quanto para o modelo de tentativa erro. No primeiro caso, ele permite que estudantes que tenham perdido a aula demonstrativa possam ter acesso a essas instruções: *“eu passava os tutoriais para eles, e que não vinha na aula tinha a oportunidade de fazer”*. Também permite eliminar a necessidade dessa demonstração ou fornecer conteúdos extras para fora do horário de aula: *“Aí normalmente eu indico vídeos para eles irem assistindo em casa e fazendo. E também se eles têm uma dúvida, porque eles não me têm do lado deles”*. Já para o modelo de tentativa e erro, os tutoriais disponíveis na internet são uma das principais fontes de informação para a descoberta autônoma de como resolver os desafios lançados pelos professores: *“E daí depois com os tutoriais eles se viram. Eles colocam lá na web: Como fazer uma rampa? Como fazer uma escada? E aparece. Eu aprendi assim”*.

A segunda estratégia mais citada foi a **demonstração pelo professor**, mencionada por onze professores, que consiste basicamente em abrir o software na frente dos estudantes, explicar suas ferramentas e funcionamento e desenvolver alguma atividade prática com ele. Essa atividade pode ser um exercício que tenha que ser feito pela turma posteriormente ou alguma explicação que depois possa ser usada de forma mais criativa no desenvolvimento do projeto. É possível que os estudantes só prestem atenção, ou que vão já seguindo o professor: *“Eu abro o software e vou mostrando passo a passo e eles vão me seguindo passo a passo”*.

A principal crítica que surgiu a essa estratégia é que ela pode ser tornar cansativa e dispersar a atenção da turma, principalmente se utilizada por muito tempo: *“muita pouca gente presta atenção, mas eu não sei como começar sem fazer isso”* ou *“é uma aula bem maçante, eles não aguentam muito tempo”*. Porém, principalmente os professores que aderem ao modelo tentativa e erro, defendem que uma demonstração inicial é importante para mostrar o funcionamento básico e dar um ponto de partida mais consistente para a exploração autônoma: *“eu dou uma aula para eles de introdução, para eles perderem o medo, mostro o básico, alguns acompanham no computador, outros não. E aí, eles perdem o medo de*

encarar o software e daí eu fico a disposição para tirar dúvidas e acompanhar no desenvolvimento do projeto”.

Também citada por onze professores está a **orientação individual**. Como o nome já indica, ela consiste no acompanhamento no uso do software, verificando se a utilização está correta e se há dúvidas a serem sanadas: *“aí eu vou indo um por um, porque eles não absorvem só porque eu falo, aí enquanto eles vão mexendo eles vão tendo as mesas dúvidas, aí eu vou mostrando direto no computador. (...) O que eu vi que funciona melhor e é o mais trabalhoso é esse um para um, eu passo ali as coisas, alguns prestam atenção e conseguem fazer sozinhos, outros não, e aqueles que realmente não aprendem eu sento do lado e vou mostrando”.* Nos modelos tutorados, a orientação acaba sendo mais intensiva, um por um, enquanto nos modelos de tentativa e erro os docentes se colocam à disposição e esperam pelos estudantes.

O **ensino por projeto** é uma estratégia comum e central nos cursos de design e foi citada também onze vezes, sendo apontada tanto pelos professores que trabalham em disciplinas isoladas de representação quanto pelos professores de disciplinas de projeto. Nas disciplinas de projeto a sua adoção é previsível, e funciona como uma forma indireta de ensino de software, já que estes acabam demandados pelos estudantes para a realização dos projetos, mesmo com pouca ou nenhuma orientação dos professores. Já nas disciplinas de representação é relevante apontar as justificativas dadas para o seu uso, que giram em torno da tentativa de conferir um propósito para o ensino e aprendizagem dos conhecimentos em representação: *“eu tenho pensado a partir de representação gráfica, seja a representação gráfica digital ou analógica, eu vejo que o ensino da ferramenta tem que estar conectado com um propósito, e o propósito deve ser o projeto, não um propósito que seja só técnico”.* Essa fala já começa a apontar para uma preocupação muito próxima a um dos interesses que motiva esta pesquisa, reforçando também nos discursos dos professores essa necessidade de contextualizar a aprendizagem da ferramenta.

Outra atividade prática muito citada foram os **exercícios práticos**, utilizados por nove docentes. Sua realização costuma focar no aprendizado de uma técnica ou ferramenta específica, preparando para a realização posterior de um projeto maior. Para auxiliar na sua realização os professores podem realizar aulas demonstrativas ou fornecer tutoriais no modelo tutorado, já no modelo tentativa e erro cabe ao estudante descobrir formas de resolvê-lo sem seguir instruções pré-determinadas.

Citada oito vezes, a **educação por pares** consiste basicamente em incentivar os estudantes a se ajudarem entre si, inclusive definindo cargos de monitores para aqueles que já tinham um conhecimento um pouco mais avançado na ferramenta: *“muitos alunos chegam com um domínio de software excepcional, aí normalmente eu conversava com esses alunos e meio que sugeria que eles ajudassem a atender toda a turma, mais ou menos como se fosse uma monitoria, porque a gente as vezes tem 30 alunos na sala. Alguns alunos gostavam, porque eles se sentiam importantes, porque se sentiam participando da aula”.* Mas essa ajuda não se limita aos casos onde um estudante tem um conhecimento maior da ferramenta, podendo ocorrer sempre: *“muita gente está falando disso, do aprender em conjunto. Porque é assim que a gente aprende, então eu sempre falo para eles, isso aqui é uma construção de conhecimentos, não sou eu que sei tudo e vocês não sabem nada, todo mundo se ajuda para tentar entender alguma coisa, então se vocês souberem falem, compartilhem isso, porque pode ajudar o outro”.*

Por fim, a **tentativa e erro** também é uma forma de desenvolver as atividades defendida por seis professores, e pode ser adotada tanto para a realização de exercícios

quanto para o desenvolvimento dos projetos. A sua adoção é o que caracteriza o modelo de mesmo nome e que se opõe ao modelo tutorado. Suas vantagens e desvantagens já foram debatidas quando discutimos os dois modelos, mas são resumidas de forma bem objetiva pela fala de uma professora: *“A tentativa e erro pode ser frustrante? Pode, e pode ter várias consequências, pode ser bom, porque a pessoa pode desenvolver uma autonomia ou autodidatismo, vai depender do perfil da pessoa, tem pessoa que vai tentar três vezes e vai se frustrar e desistir do curso, pintando cenários extremos”*.

Citadas apenas uma vez cada, estavam a utilização de **provas teóricas** e a **demonstração por estudantes** – quanto os próprios estudantes demonstram o uso do software para ensinar o restante da turma.

4.5. Avaliação da Aprendizagem

Quanto à avaliação, ficou claro nos depoimentos que apenas os professores de disciplinas cuja representação digital e o software são o conteúdo principal avaliam esses conhecimentos. Nas disciplinas de projeto ou de outras áreas no qual a representação digital não é o foco principal, os professores foram unânimes em afirmar que dão pouca atenção ao uso da ferramenta no momento de avaliar as atividades: *“Só o design, justamente porque não é uma disciplina de software”*. É interessante notar como essa frase reforça e deixa evidente a ainda forte separação entre o ensino-aprendizagem de representação digital e o restante dos conhecimentos em design já apontada por Spitz (1995) desde o final do século passado. Como esse relato aponta, existem "outras coisas" que são design e que merecem ser avaliadas em uma disciplina de projeto, mas o uso da representação digital aparentemente não é uma delas.

Entre os professores de disciplinas específicas de representação digital, cinco apontaram a avaliação *a posteriori* como a única utilizada. Essa estratégia consiste em recolher os arquivos digitais, avaliá-los quanto ao uso adequado da ferramenta e depois retornar uma avaliação aos estudantes.

Na avaliação em disciplinas específicas de representação digital, é possível que outros aspectos relacionados ao design sejam avaliados. Porém, foi comum entre os professores a afirmação de que estes outros aspectos tem um peso menor na avaliação ou até mesmo nem são avaliados: *“Eu julgava, mas eu dava um peso bem menor para as questões de design e estética do que para a técnica e organização dos arquivos”*. Essa valorização maior do software gera situações onde trabalhos ruins do ponto de vista geral do design tenham notas maiores só por um suposto uso adequado da ferramenta, e vice-versa: *“era até engraçado, que alguns trabalhos que tinham design melhor acabavam tirando nota mais baixa que outros que tinham design pior. O aluno fazia aquele projeto bonito com uma estética muito boa, mas não tratava direito a estrutura de arquivo, não usava bem as técnicas”*. Novamente é possível perceber na fala dos professores uma clara distinção entre algo que seria “o design” do qual a representação digital é algo a parte.

4.6. Necessidades de Mudanças No Ensino-Aprendizagem Atual

Referente às mudanças no ensino de representação digital que os professores gostariam de implementar, as falas giraram em torno de três temas: **umentar a responsabilidade do estudante** com o seu próprio aprendizado, **mudar a forma de inserção nos currículos** e promover uma maior **integração e contextualização com outros conhecimentos** de design.

Sete professores defendem que é preciso incentivar uma **maior autonomia do**

estudante e torná-lo mais responsável pelo seu próprio aprendizado. Nesse sentido, o papel da universidade deveria ser de apenas de mostrar caminhos e “sensibilizar” o estudante sobre esse tema, para que este partisse então para desenvolver de forma mais independente seu aprendizado em representação digital. Ou, nas palavras de uma professora: *“Sensibilizar que isso existe, e que essa possibilidade está lá, a destreza e a profundidade vai depender de cada um”*. Nesse contexto, alguns professores defendem que as disciplinas voltadas para o ensino de representação digital poderiam continuar existindo, *“mas eu acho que talvez só com aquela imersão inicial para a pessoa não ficar com medo de usar. Mas a partir desse momento que a pessoa teve a iniciação no software ela não precisa mais, porque ela tem youtube, tem tutoriais, tem vídeos”*.

Essa maior autonomia do estudante poderia ser apoiada com a mudança na forma como representação digital é inserida nos currículos. Seis professores defenderam que o ensino pode ocorrer através de **outros modelos além de disciplinas obrigatórias convencionais**, como oficinas, cursos de curta duração ou ateliês permanentes voltados para isso, todos, porém, com participação facultativa do estudante. Sobre os cursos de curta duração, uma professora defendeu: *“Cada semestre eu tenho minha grade curricular, aí para cada disciplina é indicado que você saiba tais softwares, aí você faz um workshop voltado só para o software. Aí em vez de ficar o semestre inteiro nisso, em três noites eu sensibilizo a pessoa”*. Outra forma de trabalhar com isso, em vez de cursos curtos, seriam ateliês abertos e permanentes para estudo de representação digital: *“talvez se a gente pudesse trabalhar com a ideia de oficina, fora do horário, tipo tem um monitor, um cara que manja muito de software, pro aluno ir lá desenvolver naquele laboratório, junto com o monitor, aí se aproxima mais como uma agência, um monte de gente separada em salas ou em uma salona só, e galera toda trabalhando ali e de repente ‘o fulano, como é que resolve tal coisa?’”*.

Outra mudança curricular defendida por cinco professores foi a **extinção de disciplinas isoladas** voltadas apenas para aprendizagem de software e representação digital: *“não faz o menor sentido, é perda de tempo. Porém o uso da ferramenta não pode de jeito nenhum ser deixado de fora”* ou *“sou completamente contra o ensino de desenho destacado do projeto. As disciplinas que ensinam software sem um escopo dão foco no software, sem falar como o software te ajuda a projetar, não tem como cindir uma coisa da outra”*. O que essas falas apontam é a contrariedade à existência de disciplinas isoladas e focadas apenas no software, mas não são contrárias ao ensino destes conhecimentos, que poderia migrar para um modelo de oficinas – como já mencionado anteriormente – ou ser integrado junto a discussões de projeto ou de outros conhecimentos de design.

Essa **integração e contextualização maior** entre representação digital e outros conhecimentos de design foi o terceiro tema identificado nas falas dos professores, estando diretamente ligado com o problema específico abordado na pesquisa maior à qual este trabalho está ligada. Vale notar, que até este momento, durante as entrevistas, esse problema não havia sido colocado pelo pesquisador, portanto são falas que surgiram espontaneamente dos respondentes. Para dez professores, é preciso encontrar formas para que a aprendizagem de software e representação digital ocorra de forma mais próxima, integrada e contextualizada com o restante dos conteúdos abordados no curso. Para sete professores, essa maior integração deve ocorrer através da prática projetual: *“acho que curso de design deveria ter uma disciplina gigante de projeto”, “não se pode ensinar desenho sem ter um exercício projetual por trás. Porque é muito melhor para quem está aprendendo. Você só aprende software quando você precisou fazer alguma coisa que você queria e precisava fazer”* ou *“o que eu acho que acontece com o software, e as vezes é difícil, a gente tenta fazer, mas é*

superficial, é essa parte de interdisciplinaridade. Eu estou modelando um projeto 3D, eles têm que modelar o que eles tão fazendo na disciplina de projeto". Mais sobre esse tema será abordado no tópico a seguir.

4.7. Opiniões Sobre o Ensino-Aprendizagem Contextualizado

Seguindo o roteiro da entrevista, o último tema abordado justamente indagava os professores sobre o que achavam do desenvolvimento de novos modelos de ensino que buscassem integrar e contextualizar melhor representação digital e outros conhecimentos em design. Além dos dez professores que já haviam apontado a importância de mudanças em busca dessa maior integração, os seis demais respondentes também concordaram com a necessidade dela: *"É o meu sonho, acho que faz muito sentido. Eu tenho sentido com os testes que tenho feito que faz muito mais sentido, é estranho ser separado"*. Porém, além de apresentarem defesas dessa mudança, também foram feitas algumas ressalvas. Ambos os pontos, positivos e negativos, são abordados a seguir.

A defesa da contextualização apontou que esta pode melhorar tanto o ensino específico de representação digital quanto a formação em design como um todo por quatro motivos: conferir propósito ao ensino de representação digital; reforçar claramente o papel do software como ferramenta, como meio e não fim; aumentar o engajamento dos estudantes e permitir a diminuição da carga horária dos cursos.

A palavra **propósito** especificamente foi utilizada por três professores para justificar a importância da contextualização: *"que deixa o aprendizado do software mais fácil é eles verem o propósito do que estão fazendo"* ou *"o ensino da ferramenta tem que estar conectado com um propósito, e o propósito deve ser o projeto, não um propósito que seja só técnico"*. Outros termos similares também foram usados por outros professores, como por exemplo "sentido" e "objetivo maior". O que essas falam apontam é que o aprendizado de representação digital que realmente contribua para a formação e a atuação do designer é potencializado quando já ocorre conectado aos outros conhecimentos em design e à prática projetual. Como afirma uma professora: *"Para mim tem que ser através do projeto [...] Precisa ter um sentido para eles verem a necessidade de fazer a representação, [...] e aí a representação pela representação não leva para o projeto"*. De forma similar, outra professora defende que *"você só aprende software quando você precisa fazer alguma coisa que você queria e precisava fazer. Então essa coisa de tutorial, de passo, de desmontar um negócio para desenhar a peça mecânica não faz sentido antes. Tentativa e erro e ter um objetivo concreto, e um tema pelo qual cada estudante se apaixona e quer desenvolver, faz com que eles interiorizem a lógica do instrumento representativo"*.

A adição de propósito ao ensino e a aproximação do software com outros temas também pode – segundo alguns professores – **melhorar o engajamento dos estudantes**: *"O ponto positivo que eu vejo é que tende a engajar mais os alunos, porque às vezes você fica com exercícios mais abstratos o aluno olha para aquilo e pensa 'para que serve, quando vou usar isso?' Então acho que você pode trabalhar mais buscando essas aplicações, e pensando na relação professor aluno, é uma coisa que dá menos margem pro aluno ficar reclamando 'professor, não vejo finalidade nisso, isso não vai servir pra nada'"*.

Outro ponto apresentando por professores é a que a integração e contextualização do ensino de representação digital é importante para que o aluno compreenda o seu **papel como ferramenta** da prática do design, como meio e não fim, diminuindo a chance de este supervalorizar a ferramenta ou achar que apenas ela é suficiente. Como afirmou um professor,

essa contextualização pode ajudar os estudantes a verem “*que aquilo é uma ferramenta, isso aqui não é design. Você saber mexer nesse software é tipo ‘você é um bom mexedor de software’, mas você não é um designer*”.

Para alguns professores, existe, portanto, um risco do ensino isolado de software formar o que eles pejorativamente chamam de ‘micreiros’, em uma analogia similar aos ‘macacos de computador’ citada por Stout *apud* Marshall & Meachem (2007). Um estudante ou profissional neste perfil confunde a prática da representação digital com a própria prática do projeto de design como um todo, como se a sua habilidade em representar algo no computador fosse o elemento mais importante para o desenvolvimento do projeto. Ou, como afirma Stout, serem muito bons em usar as tecnologias, mas deixando o ‘rabo balançar o cachorro’, já que ao não terem um entendimento claro de como organizar textos e elementos para criar uma mensagem eficaz, deixam o computador cuidar disso por eles.

Neste sentido, uma professora questionou: “*Como que faz para ensinar software sem formar um ‘micreiro’? Para mim tem que ser através do projeto, igual ensinar desenho. Precisa ter um sentido para eles verem a necessidade de fazer a representação, já que o software é representação também. E aí a representação pela representação não leva para o projeto. A chance de ela querer copiar alguém é maior. Acho que o software pode condicionar a maneira de pensar, e pode dar a falsa sensação de que a pessoa sabe fazer projeto* [grifo do autor].”

Na mesma linha, outra professora apontou: “*No caso do design editorial, o mais importante é desenvolver essa sensibilização do olhar, depois ferramenta ele aprende, o contrário não acontece, aprender o software e achar que vai fazer um bom livro, não vai acontecer, é só uma ferramenta*”.

Por fim, a última vantagem apontada por dois professores seria a possibilidade de redução da carga horária com disciplinas focadas apenas no uso da ferramenta, liberando professores e estudantes para outras atividades que seriam mais valorosas. Porém, esse ponto, apontado como vantagem por alguns professores, também é alvo de ressalvas por outros. Como afirmam outros dois professores, discutir representação digital junto com projeto pode reduzir além do adequado o tempo dedicado à prática projetual: “*O software vai tirar um pouco de tempo de projeto e projeto talvez acabe ficando um pouco de lado*”, “*desse jeito tem um lugar que é muito bom que você contextualiza tudo, mas também tem que dar tempo de eles projetarem*”.

Outra ressalva colocada por professores é sobre o perigo que se corre de “*querer ensinar tudo ao mesmo tempo e acabar não ensinando nada direito*”. Ou seja, os momentos de aprendizado e prática integrados são importantes sim, mas também devem ser dosados com olhares e discussões sobre pontos específicos: “*tudo junto ao mesmo tempo não funciona, tem que ter uma divisão de prioridades, entender complexidade e tal, e talvez ter momentos de vamos abrir esse pedacinho aqui e estudar mais e depois volta para o contexto*”.

A terceira e uma última ressalva apontada reflete sobre os riscos que uma vinculação muito forte do ensino de representação digital com a prática projetual pode provocar. Como alguns professores colocam, o software condiciona a prática do projeto e ser for utilizado como única ferramenta de representação vai gerar limitações: “*O importante é despertar a autonomia deles para irem atrás das coisas. Não acho que seja tão importante ensinar a ferramenta, e eu prefiro não condicionar a ferramenta porque ela condiciona o resultado*”. Nesse sentido, é importante que o software seja integrado e contextualizado junto com a prática projetual, porém sem substituir outras formas de representação, sendo mais uma opção: “*Acho muito importante essa passagem entre os meios de representação bi e tri. Faz*

um desenho, vai pro software, faz um desenho digital 3D, dai vai pra modelagem, dai volta pro desenho. Esse é meu sonho de mudar, é isso, essa conexão entre as linguagens. Para entender que é linguagem. Em vez de ter um estudo isolado das técnicas, ter um projeto que permita transitar entre as várias linguagens, e entender o que cada uma me dá, que limitações que ela tem, que potencialidades que têm”.

Sendo coerente com suas defesas, seis professores relataram alguma forma de tentativa de contextualização entre conhecimentos de representação digital e outros conhecimentos de design. Três professoras de projeto que trabalharam em algum momento de suas disciplinas com projetos de design editorial relataram que deram aulas introdutórias no uso do software Adobe InDesign, mesmo com este conteúdo não estando previsto na ementa da disciplina. Elas também se colocavam a disposição para auxiliar em dúvidas na utilização do software. Duas professoras de representação, cujas disciplinas eram voltadas para técnicas mais manuais, também informaram que buscaram trazer o trabalho com software para dentro de suas aulas, desenvolvendo um exercício de elementos plásticos tridimensionais que iniciava com técnicas manuais e terminava utilizando o software. Uma professora de representação digital afirmou que avalia o design dos trabalhos tanto quanto o uso da ferramenta, e que também busca fazer trabalhos em conjunto com outras disciplinas. Por fim, outra professora de representação digital afirmou que busca sempre contextualizar em suas explicações para que serve cada ferramenta na prática do design, e que na sua avaliação sempre busca avaliar tanto o design quanto o uso da ferramenta.

Porém, nenhum docente conseguiu descrever um modelo no qual a integração e contextualização sejam o objetivo principal e não um acontecimento pontual em suas disciplinas. Além disso, principalmente os **professores de representação digital**, afirmam que em vários projetos **o foco não está em discutir ou avaliar o design**, mas sim ensinar a ferramenta. Do lado oposto, **professores de projeto apontam que pouco abordam e nem avaliam o uso da ferramenta**, apenas o resultado do projeto, também não fornecem material didático ou exercícios sobre isso, apenas uma aula introdutória e depois se colocam a disposição para dúvidas.

5. Considerações Finais

O objetivo deste trabalho foi coletar e analisar as opiniões de docentes sobre os processos de ensino-aprendizagem de software e representação digital em cursos superiores de design no Brasil. A pesquisa partiu de uma abordagem qualitativa, mais interessada em aprofundar os entendimentos sobre as opiniões encontradas do que quantificá-las de forma estatística e generalizável.

A partir dos resultados obtidos, considere-se que o objetivo foi cumprido, já que os dados coletados com dezesseis docentes de seis diferentes universidades públicas brasileiras permitiram traçar um quadro rico de opiniões em comum, assim como de opiniões divergentes. Além do agrupamento destas opiniões em temas que puderam ser mais facilmente explicados, a utilização de trechos relevantes de falas dos participantes permitiu um entendimento mais profundo e qualitativo desses temas.

Quanto à importância de se possuir conhecimentos em software e representação digital para a prática do design, todos os participantes afirmaram que está é alta ou muito alta. A justificativa dada é que esses conhecimentos são demandados pelo mercado e que são fundamentais para a prática projetual.

Essa alta importância reflete na necessidade de designers já saírem com esses conhecimentos de seus cursos superiores, o que leva ao segundo tema discutido, de quem seria a responsabilidade pelo processo de aprendizagem: iniciativa própria dos estudantes ou um processo guiado pelos docentes e pela instituição? Todos os entrevistados de certa forma apontaram que é uma responsabilidade compartilhada. Porém, nas explicações sobre esse compartilhamento, ficou claro pelo menos dois pólos antagônicos de possibilidades: (1) a responsabilidade é maior do estudante, que deve ter autonomia e iniciativa de “correr atrás” desses conhecimentos, cabendo à instituição apenas apontar alguns caminhos iniciais, orientar quando necessário e cobrar essa aprendizagem; (2) de maneira oposta, cabe a instituição guiar a maior parte do processo, através de disciplinas focadas nesse tema, aulas expositivas, realização de exercícios e acompanhamento mais próximo dos docentes, mas ainda assim, esse ensino não seria suficiente, cabendo ao estudante ter a iniciativa de se aprofundar, principalmente nos software e áreas de seu maior interesse.

Refletindo esses dois pólos antagônicos, foi possível também encontrar dois modelos de ensino de software e representação digital. O modelo tutorado é aquele relacionado com a visão de que é grande o papel dos professores e da universidade nesse processo, cabendo estruturar de forma mais presente e objetiva esse ensino, através de tutorais, demonstrações, exercícios e avaliações focadas nesse conhecimento de forma mais constante e focada. De forma oposta, o modelo da tentativa e erro coloca a maior parte da responsabilidade na iniciativa própria dos estudantes, cabendo ao professor lançar problemas e avaliar os resultados, mas sem dar um passo a passo, que deve ser descoberto pelos estudantes de forma autônoma. Apesar de opostos, é possível trabalhar de forma equilibrada com ambos os modelos, aplicando-os em momentos diferentes do processo, o que neste trabalho foi denominado de “caminho do meio”, conforme a fala de uma professora.

Sobre as mudanças que os participantes consideraram necessárias, três recomendações foram identificadas: (1) aumentar a responsabilidade do estudante no processo; (2) alterar as formas de inserção deste conteúdo nos currículos; e (3) integrar e contextualizar mais o ensino-aprendizagem de software e representação digital com outras áreas do design, principalmente à prática projetual.

Essa última recomendação de mudança é um tema de interesse central nesta pesquisa, e por isso foi debatido mais a fundo com os participantes. Todos concordaram que é importante encontrar formas de aumentar a integração e contextualização do ensino-aprendizagem com outros conhecimentos em design. Os motivos para isso seriam: conferir “propósito” e “sentido” ao ensino de representação digital; reforçar o papel do software como ferramenta, como meio e não fim; aumentar o engajamento dos estudantes e permitir a diminuição da carga horária dos cursos. Porém, também foram feitas algumas ressalvas sobre problemas que podem surgir nessa integração, como o risco de acabar reduzindo demais o tempo destinado ao aprendizado de projeto ou de representação digital, “querer ensinar tudo ao mesmo e não ensinar nada” ou de vincular demais as práticas projetuais ao uso de software acabando por eliminar outras possibilidades de representação.

Ao término desta pesquisa, o que se percebe é que existe um reconhecimento da importância em integrar e contextualizar o ensino-aprendizagem de software e representação digital com outros conhecimentos e práticas do design. Porém, esse reconhecimento pouco se traduz em práticas efetivas que permitam isso. Os conhecimentos ainda tendem a ser abordados de forma muito isolada como Spitz (1995) já havia notado desde o final do século passado. Tantos anos depois, as disciplinas de projeto ainda dão pouca atenção para software e representação digital, abordando o tema em momentos muito pontuais ou ignorando-o por

completo. Nas disciplinas de representação digital, existe em alguns casos uma tentativa maior de integração, como por exemplo através da proposição de projetos de design, porém, como ficou claro, mesmo nesses casos pouco se discute sobre o projeto, com a avaliação focando apenas ou principalmente no uso das ferramentas de representação.

A partir desse diagnóstico, o que se desdobra para pesquisas futuras é justamente a busca pela proposição de estratégias mais efetivas para a implementação dessa integração e contextualização do ensino-aprendizagem de software e representação digital. De modo que essa vontade já expressada pelos docentes possa de fato se tornar uma realidade. E que representação digital não seja mais vista como algo isolado, separada do design e menos importante de um lado, assim como não seja mais supervalorizada e confundida com a própria prática de design por outro, mas sim como uma das facetas que torna o fazer do design completo.

Referências

- ALDOY, N. **An investigation into a digital strategy for industrial design education**. 2011. Tese (Doutorado) – Loughborough University, Loughborough, 2011.
- ATHARIFAR, H. YILDIZ, F. & KNAPP, J. R. Survey of the Current Academic and Industrial Trends in Utilizing the CADD Technology. **120th ASEE Annual Conference & Exposition**, 2013.
- FONTANELLA, B. J. B.; LUCHESI, B. M.; SAIDEL, M. G. B.; RICAS, J.; TURATO, E. R., & MELO, D. G. Amostragem em pesquisas qualitativas: proposta de procedimentos para constatar saturação teórica. **Cadernos de Saúde Pública**, v. 27, n. 2, p. 388-394, 2011.
- GOLDSCHIMIDT, G. Design Representation: Private Process, Public Image. In: GOLDSCHIMIDT, G & PORTER, W. **Design Representation**, Londres: Springer, 2004.
- LAUER, C. High-Tech Invention: Examining the Relationship Between Technology and Idea Generation in the Document Design Process. **Journal of Business and Technical Communication**, v.29, n.4, 2015.
- LYNN, D. Automotive design education embraces the digital age. **IDSA – IDSA International Design Conference**, 2006.
- MARSHALL, L. Technological Tools: the Need to Situate Software Skills in the Implementation of Design Concepts. **IEEE International Conference on Advanced Learning Technologies (ICALT'04)**, 2004.
- MARSHALL, L. & MEACHEM, L. Direct or directed: orchestrating a more harmonious approach to teaching technology within an Art & Design Higher Education curriculum with special reference to visual communications courses. **Learning, Media and Technology**, v. 32, n.1, 2007, p. 41-52.
- MATTÉ, V. A. **O Conhecimento da prática projetual dos designers gráficos como base para o desenvolvimento de materiais didáticos impressos**. 2009. 304f. Tese (Doutorado) – Programa de Pós-Graduação em Engenharia e Gestão do Conhecimento – Universidade Federal de Santa Catarina, Florianópolis, 2009.

MAZZAROTTO, M. & ULBRICHT, V. O ensino formal de softwares em cursos de Design Gráfico: uma presença fragmentada. **12º Congresso Brasileiro de Pesquisa e Desenvolvimento em Design**. Belo Horizonte, v.2, n.9, novembro 2016.

MILES, M. B.; HUBERMAN, A. M. & SALDAÑA, J. **Qualitative data analysis: a methods sourcebook**. London: Sage Publications, 2014.

PINTO, A. C. Aprender a aprender o quê? Conteúdos e estratégias. **Psicologia, Educação e Cultura**, v. 2, n.1, 1996, p. 37-53.

ROSSELLI, B. **Taxonomia de meios de representação em ambientes multidimensionais e sua aplicação na metodologia projetual**. Dissertação (Mestrado) – Mestrado em Design – Centro Universitário Ritter dos Reis, 2012.

SPITZ, R. Dirty hands on the keyboard. **Sixth International Symposium on Electronic Art - ISEA'95**, Montreal, 1995.

YANG, M. Y., YOU, M., & CHEN, F. C. Competencies and qualifications for industrial design jobs: Implications for design practice, education, and student career guidance. **Design Studies**, v.26, n.2, 2005, p. 155–189.